

گزارش کامپیوتر

سال سی و چهارم، خرداد و تیر ۱۳۹۱، شماره ۲۰۳، ۲۰۰۰ تومان

مقاله

- ۸ سامانه‌ای به کمک رایانه برای خدمات حسابداری مدیریت - سید ابراهیم ابطی
- ۱۴ محاسبات کوانتومی: از نظریه تازبان‌ها - مصطفی برمشوری، محمد هادی منصوری
- ۲۴ تحقق شهر الکترونیکی - اردوان مجیدی، رضا درخشان
- ۵۳ مروری بر روش‌های جستجو در داده‌های رمز شده - لیلا معظمی گودرزی، مجید غیوری
- ۶۸ ارائه یک پروتکل امنیت مسیر یابی در شبکه‌های موردی - علی یزدانیان، سیامک لونی
- ۷۶ انبار داده و پردازش تحلیلی بر خط - داود صالحی

گزارش

- ۱۲ بیانیه منابع آموزشی آزاد - فریده مشایخ
- ۲۳ شانزدهمین همایش بین‌المللی هوش مصنوعی
- ۳۵ بررسی عملکرد انجمن‌های علمی کشور
- ۷۴ سمینار مروری بر جنبه‌های بارز ویرایش جدید C++
- ۷۵ سمینار صنعت بازی‌های رایانه‌ای

بخش‌ها

- ۳۸ خواندنی - برنامه‌نویسان در حال کار (قسمت سیزدهم) - ابراهیم نقیب‌زاده مشایخ
- ۶۳ دیدگاه - به یاد معلمی که شاگردانش او را استاد می‌نامند - سید ابراهیم ابطی
- ۸۶ کتاب‌شناسی - جهان رقمی آینده - سید ابراهیم ابطی
- ۹۴ واژه‌گزینی - معادل فارسی گرید - سیامک کاظمی
- ۹۵ گوناگون - رتبه‌بندی دانشگاه‌های برتر جهان - حمید رضا آراسته، سید طاهر فاضلی
- ۱۰۲ سرگرمی - یک تقسیم بر صفر

اخبار

- ۲ اخبار انجمن - اخبار شاخه‌های دانشجویی - اخبار اعضای حقوقی
- ۴ اخبار ایران
- ۴ اخبار جهان

اعضای هیأت تحریریه و ویراستاران علمی

- مهندس سید ابراهیم ابطی
- دکتر مجید علیزاده
- دکتر کامبیز بدیع
- دکتر محسن رستمی
- دکتر محمد صنعتی
- دکتر عباس نوذری دالینی
- مهندس سعید امامی
- دکتر لطف‌علی بخشی
- مهندس علی پارسا
- دکتر محسن صدیقی مشکنانی
- دکتر اسلام ناظمی
- دکتر محمد گنج تابش



صاحب امتیاز: انجمن انفورماتیک ایران

مدیر مسئول و سردبیر: ابراهیم نقیب‌زاده مشایخ

مقاله‌ها و گزارشهای تالیفی یا ترجمه خود را برای گزارش کامپیوتر بفرستید. مطالب ارسالی را با خط خوانا (با ماشین شده) بر یک روی کاغذ بنویسید فاصله کافی برای افزودن اصلاحات ویرایشی در بین خط‌ها و حاشیه در نظر بگیرید. در صورت ارسال مطلب ترجمه شده، یک نسخه از اصل مطلب را نیز ارسال دارید. شکلها باید با روان‌نویس مشکی ترسیم و به‌صورت آماده چاپ ارسال شوند. مطالب ارسالی پس فرستاده نمی‌شوند.

مسئولیت مطالب گزارش کامپیوتر با نویسندگان است و لزوماً نشان‌دهنده نظر انجمن انفورماتیک ایران نیست. ذکر نام شرکتها و محصولات در مطالب گزارش کامپیوتر برای ارائه اطلاع به خوانندگان است و نباید به معنی تأیید انجمن از آنها تلقی شود.

تمام حقوق به انجمن انفورماتیک ایران متعلق است. نقل مطالب گزارش کامپیوتر با ذکر منبع بلامانع است.

سازمان آگهی‌ها: مهناز خاوندکار

تلفکس: ۸۸۳۲۸۴۱۷-۲۱

حروفچینی: تارتن سامانه

لیتوگرافی: نگارین پرتو ۷۷۵۳۰۳۰۷

چاپ علوی تهران: ۶۶۷۰۱۵۲۷

خ جمهوری، ۳۰ تیر، کوچه مصری پور، پلاک ۲۵۶

اخبار انجمن

استقبال از شماره ۲۰۲

شماره ۲۰۲ نشریه گزارش کامپیوتر که از طرف گروه تخصصی راهبری و مدیریت خدمات فناوری اطلاعات انجمن تهیه گردیده بود و به مباحث و مطالب مختلف این حوزه اختصاص داشت با استقبال زیادی از سوی نهادها و سازمان‌های آموزشی دولتی و خصوصی و همچنین کارشناسان و شرکت‌های رایانه‌ای قرار گرفت. این استقبال و درخواست نسخه‌های اضافی به حدی بود که یکماه پس از انتشار، کلیه نسخه‌های چاپ شده آن، به استثنای اندکی که از هر شماره در دفتر مجله نگاهداری می‌شود، به فروش رسید و بسیاری از علاقه‌مندان موفق به دریافت این نشریه نشدند.

گروه تخصصی راهبری و مدیریت خدمات فناوری اطلاعات، چاپ دومین ویژه‌نامه را در دستور کار خود قرار داده است.

برگزاری وبینار از طریق وبگاه انجمن

هیأت مدیره انجمن تصویب کرد که بخشی از

سخنرانی‌های ماهانه انجمن به صورت وبینار آماده گردد و از طریق وبگاه انجمن در اختیار علاقه‌مندان قرار گیرد.

این تصمیم به دلیل مشکل تداخل ساعت برگزاری سخنرانی‌های ماهانه انجمن با ساعت کاری افراد و مشکلات ترافیکی گرفته شده است و در صورتی که مورد استقبال اعضای انجمن قرار گیرد، رفته‌رفته تعداد بیشتری از سخنرانی‌ها به صورت وبینار ارائه خواهد شد.

شرکت در اولین کنفرانس الگوی اسلامی ایرانی پیشرفت

اولین کنفرانس "الگوی اسلامی ایرانی پیشرفت: نقشه راه طراحی و تدوین الگو" در روزهای دهم و یازدهم خرداد ماه ۱۳۹۱ در محل کتابخانه ملی برگزار شد.

از رؤسای انجمن‌های علمی کشور برای شرکت در نشست افتتاحیه و دیگر نشست‌های کنفرانس دعوت به عمل آمده بود که آقای نقیب‌زاده مشایخ، رئیس انجمن انفورماتیک ایران نیز در نشست روز دوم کنفرانس شرکت نمود. لازم به ذکر است که موضوع نشست روز دوم

کنفرانس، تبیین نقش انجمن‌های علمی در طراحی و تدوین الگو بود.

شرکت در هم‌اندیشی مرکز ملی مطالعات جهانی شدن

کمیسیون انجمن‌های علمی ایران به منظور تعامل بیشتر انجمن‌های علمی با مراجع علمی و بین‌المللی، در تاریخ ۹۱/۳/۹ یک جلسه هم‌اندیشی با حضور رؤسای انجمن‌های علمی فعال کشور و مسئولان مرکز ملی مطالعات جهانی شدن ریاست جمهوری ترتیب داد. در این جلسه که به مدت ۴ ساعت در مجتمع فرهنگی نهاد ریاست جمهوری برگزار شد، آقای مهندس محمد حسن محوری، عضو هیأت مدیره و خزانه‌دار انجمن انفورماتیک ایران به نمایندگی از این انجمن شرکت نمود. در این جلسه، تفاهم‌نامه همکاری بین مرکز ملی مطالعات جهانی شدن و شورای انجمن‌های علمی کشور به امضاء رسید.

شرکت در همایش بازنگری برنامه‌های درسی

گروه برنامه‌ریزی فنی و مهندسی وزارت

اول رئیس جمهوری و وزیر ارتباطات برگزار شد، موفق به کسب رتبه نخست کارگروه کارآفرینی و توسعه کسب و کار شد. لازم به یادآوری است که در این جشنواره، هشت کارگروه:

- تولید و پشتیبانی
 - پژوهش و نوآوری
 - کیفیت و ارائه خدمات شبکه‌ای
 - آموزش و اطلاع‌رسانی
 - کارآفرینی و توسعه کسب و کار
 - ارائه خدمات محتوا
 - توسعه صادرات حوزه فناوری اطلاعات
 - جایزه ملی کیفیت
- تشکیل شده بود که از میان ۹۰۰ طرح ارائه شده، برترین‌های هر گروه انتخاب شدند.

مسابقه مقاله‌نویسی درباره «محاسبات ابری»

شرکت رهنمون فناوری اطلاعات (سهامی خاص)، مسابقه مقاله‌نویسی حول محور «محاسبات ابری» را برگزار می‌نماید. کلیه علاقه‌مندان می‌توانند مقالات خود را تا تاریخ ۱۳۹۱/۵/۳۱ به وبگاه این شرکت به نشانی www.rahnamoonco.com ارسال نمایند. اسامی برندگان در تاریخ ۱۳۹۱/۶/۱۴ همزمان با سالروز تولد ابوریحان بیرونی اعلام خواهد گردید و مراسم اهداء جوایز در دی ماه همزمان با دهمین سالگرد فعالیت این شرکت برگزار خواهد شد.

- موضوع مقالات عبارتند از:
- تاریخچه محاسبات ابری
 - کاربردهای محاسبات ابری در جوامع امروز
 - تجربیات موفق محاسبات ابری در دنیا
 - زیرساخت‌های مورد نیاز محاسبات ابری
 - چالش‌های ارائه‌دهندگان خدمات محاسبات ابری
 - طرح پیشنهادی برای پروژه کاربردی محاسبات ابری در ایران
- مقالات باید به زبان فارسی نوشته شده باشند و مقالات برتر به صورت یک مجموعه چاپ و منتشر خواهد شد.

اخبار شاخه‌های دانشجویی

برگزاری سخنرانی علمی در دانشگاه تهران

شاخه دانشجویی انجمن در دانشکده ریاضی، آمار و علوم کامپیوتر دانشگاه تهران، یک سخنرانی علمی در تاریخ ۲۴ اردیبهشت ۹۱ برگزار کرد. موضوع این سخنرانی "مروری بر جنبه‌های بارز ویرایش جدید زبان برنامه‌نویسی ++C" بود که توسط آقای مهندس سعید امراللهی بیوکی انجام گرفت. این سخنرانی با استقبال خوبی از سوی دانشجویان روبرو شد و تعدادی از اعضای هیأت علمی دانشکده نیز در آن حضور داشتند.

گزارش مفصل‌تری از برگزاری این سخنرانی در همین شماره نشریه درج شده است.

برگزاری سخنرانی علمی در دانشگاه شهید بهشتی

شاخه دانشجویی انجمن در دانشکده مهندسی برق و کامپیوتر دانشگاه شهید بهشتی، یک سخنرانی علمی در تاریخ ۹۱/۲/۲۵ در محل این دانشکده برگزار کرد. در این جلسه آقای مهندس فرزاد ملک‌آرا درباره صنعت بازی‌های رایانه‌ای سخنرانی کرد.

این سخنرانی با استقبال زیادی از سوی دانشجویان روبرو گشت. تعدادی از دانشجویان دانشکده علوم کامپیوتر این دانشگاه نیز در این جلسه حضور یافته بودند. گزارش مفصل‌تری از برگزاری این سخنرانی در همین شماره نشریه درج شده است.

اخبار اعضای حقوقی انجمن

خبری از شرکت خدمات انفورماتیک

شرکت خدمات انفورماتیک در دومین جشنواره ملی فاوا که در تاریخ ۲۷ اردیبهشت ۹۱ به مناسبت روز جهانی ارتباطات و با حضور معاون

علوم، تحقیقات و فناوری به منظور بازنگری برنامه‌های درسی و لزوم تسهیل در ایجاد مجموعه‌های آموزشی جدید، همایش یک روزه‌ای را با حضور استادان متخصص، صاحب‌نظران و نمایندگان انجمن‌های علمی در روز سه‌شنبه ۹۱/۴/۶ در محل آمفی‌تئاتر مرکزی دانشگاه صنعتی امیر کبیر برگزار کرد.

- محورهای این همایش عبارت بودند از:
- بررسی کمی و کیفی برنامه‌های تحصیلات تکمیلی
 - استخراج شاخص‌های مناسب برای تدوین الگوی بازنگری
 - افزایش وزن پژوهش در آموزش‌های مهندسی
 - ایجاد انعطاف در برنامه‌های تحصیلات تکمیلی
 - بازنگری و بروزرسانی برنامه‌های آموزشی و پژوهشی در راستای رفع نیازهای صنعت

آقایان نقیب‌زاده مشایخ، رئیس هیأت مدیره و ابطحی، نایب رئیس و سرپرست کمیته گردهمایی‌های علمی انجمن انفورماتیک ایران در این همایش به نمایندگی از انجمن حضور داشتند.

شرکت در هم‌اندیشی سازمان ملی استاندارد

سازمان ملی استاندارد به منظور توسعه همکاری‌های ملی و تقویت همکاری با انجمن‌های علمی، یک جلسه هم‌اندیشی با حضور رئیس و معاونان سازمان ملی استاندارد و رؤسای انجمن‌های علمی فعال کشور در روز شنبه ۹۱/۴/۳ در محل سازمان اسناد و کتابخانه ملی برگزار کرد. در این جلسه، تفاهم‌نامه همکاری بین شورای انجمن‌های علمی ایران و سازمان ملی استاندارد به امضاء رسید.

لازم به ذکر است که آقای نقیب‌زاده مشایخ، رئیس انجمن انفورماتیک ایران در این جلسه هم‌اندیشی شرکت نمود.

اخبار ایران

یازدهمین کنفرانس آمار ایران

یازدهمین کنفرانس آمار ایران در تاریخ ۷ تا ۹ شهریور ۱۳۹۱ در دانشگاه علم و صنعت ایران برگزار خواهد شد. این کنفرانس از طرف انجمن آمار ایران و با همکاری دانشگاه علم و صنعت برگزار می‌گردد. علاقه‌مندان برای دریافت اطلاعات بیشتر در مورد اهداف، محورهای تخصصی، کارگاه‌ها و اخبار کنفرانس می‌توانند به وبگاه <http://isc11.iust.ac.ir> مراجعه کنند.

چهل و سومین کنفرانس ریاضی ایران

چهل و سومین کنفرانس ریاضی ایران در تاریخ ۶ تا ۹ شهریور ۱۳۹۱ در دانشگاه تبریز برگزار خواهد شد. این کنفرانس از سوی انجمن ریاضی ایران و با همکاری دانشگاه تبریز برگزار می‌گردد. علوم کامپیوتر یکی از محورهای این کنفرانس است. علاقه‌مندان برای دریافت اطلاعات بیشتر می‌توانند به وبگاه کنفرانس به نشانی <http://imc43.tabrizu.ac.ir> مراجعه کنند.

ششمین سمپوزیوم بین‌المللی مخابرات

مؤسسه تحقیقات ارتباطات و فناوری اطلاعات ششمین سمپوزیوم بین‌المللی مخابرات را از تاریخ ۱۶ تا ۱۸ آبان ماه سال جاری برگزار می‌کند. این همایش با هدف گسترش فنی و هم‌اندیشی در حوزه‌های مخابرات و کامپیوتر برگزار می‌شود و طی آن آخرین پیشرفت‌های علمی و پژوهشی، آینده فناوری‌های مخابراتی و همچنین مفاهیم و نظریه‌های جدید در قالب مقالات علمی و کارگاه‌های آموزشی و میزگردهای تخصصی ارائه خواهد شد.

محورهای سمپوزیوم عبارتند از:

- ارتباطات بیسیم، سیار و ماهواره‌ای
- شبکه‌های نوری و شبکه‌های دستیابی

• نظریه شبکه‌ها

• فناوری اطلاعات

• مسائل امنیتی در فناوری اطلاعات و ارتباطات

• تصمیم‌گیری و محاسبات

• جنبه‌های راهبردی فناوری اطلاعات و ارتباطات

علاقه‌مندان برای دریافت اطلاعات تکمیلی می‌توانند به وبگاه سمپوزیوم به نشانی <http://ist2012.itrc.ac.ir> مراجعه کنند.

انجمن انفورماتیک ایران جزء حامیان علمی این کنفرانس است.

علاقه‌مندان برای دریافت اطلاعات بیشتر می‌توانند به وبگاه کنفرانس به نشانی www.petroict.com مراجعه کنند.

دوازدهمین کنفرانس سیستم‌های فازی

انجمن سیستم‌های فازی ایران با همکاری دانشگاه مازندران، دوازدهمین کنفرانس سیستم‌های فازی ایران را از تاریخ ۲ تا ۴ آبان ماه ۱۳۹۱ در محل دانشکده علوم ریاضی آن دانشگاه برگزار می‌کند. محورهای این همایش عبارتند از:

- مبانی مجموعه‌ها و منطق
- شناسایی و پیش‌بینی سیستم‌ها
- کنترل و رباتیک
- تحقیق در عملیات
- سیستم‌های هوشمند
- اقتصاد، مدیریت و تجارت

• علوم کامپیوتر

• نظریه تصمیم

• علوم پزشکی

• علوم انسانی

علاقه‌مندان برای دریافت اطلاعات بیشتر می‌توانند به وبگاه کنفرانس به نشانی <http://icfs1391.umz.ac.ir> مراجعه کنند.

نشانی جدید وب‌نوشت استادان علیه تقلب

وب‌نوشت (بلاگ) استادان علیه تقلب که به

همت تعدادی از استادان دانشگاه‌های کشور و با هدف مبارزه با تقلب و تقلب علمی در دانشگاه‌ها، چه از سوی دانشجویان و چه از سوی اعضای هیأت علمی، به وجود آمده است، در نشانی جدید pap.blog.ir قابل دسترسی است.

با گسترش حرکت‌های غیراخلاقی در فضای علمی کشور که با انگیزه‌هایی چون اخذ مدرک، پذیرش یا ارتقاء مرتبه دانشگاهی صورت می‌گیرند، این وب‌نوشت قصد دارد برخی موارد و روش‌های تقلب را بیان و ضمن آموزش به دانشجویان و تلاش برای اشاعه اخلاق و آداب حرفه‌ای، مسئولان را وادار کند تا به مشکل تقلب و ریشه‌های آن واکنش جدی نشان دهند.

اخبار جهان

کامپیوتر لوحی شرکت مایکروسافت

شرکت مایکروسافت با این امید که بتواند بخشی از بازار بزرگ کامپیوترهای لوحی را که هم اکنون در اختیار رقیبانی، همچون اپل و سامسونگ است از آن خود کند، نمونه‌ای از نخستین کامپیوتر لوحی خود را برای گروهی از کارشناسان به نمایش گذاشت. نام این کامپیوتر لوحی که قرار است در گونه عادی و حرفه‌ای روانه بازار شود سورفیس (surface) است.

سورفیس عادی با ۹/۳ میلیمتر ضخامت و ۶۸۰ گرم وزن، از رقیب اصلیش یعنی آی‌پد اپل، یک دهم میلیمتر نازک‌تر اما ۹۰ گرم سنگین‌تر است. سیستم عامل این مدل ویندوز آرتی خواهد بود.

گونه حرفه‌ای که با سیستم عامل ویندوز ۸ ارائه خواهد شد، ۱۳/۵ میلیمتر ضخامت و ۹۱۰ گرم وزن دارد.

این کامپیوتر لوحی مجهز به کارت خوان، درگاه ارتباطی USB 2.0 (گونه عادی) و USB 3.0 (گونه حرفه‌ای) بوده و با ظرفیتی

بین ۳۲ تا ۱۲۸ گیگا بایت وارد بازار خواهد شد.

در نگاه اول آنچه به چشم می‌آید، صفحه کلید مغناطیسی و فوق‌العاده ظریف این کامپیوتر لوحی است که با تنها ۳ میلیمتر ضخامت، اگر چه بسیار شبیه پوشش هوشمند آی پد اپل به نظر می‌آید اما در واقع یک صفحه کلید کامل است و برای خیلی‌ها که با صفحات لمسی چندان راحت نیستند می‌تواند بسیار مطلوب باشد.

در گونه حرفه‌ای سورفیس، یک قلم مخصوص هم طراحی و تعبیه شده که کاربر می‌تواند به کمک آن، یادداشت‌های شخصی‌اش را با دستخط خود بنویسد و ذخیره کند. با استفاده از این قلم، علاوه بر طراحی و نقاشی و رسم نمودار، می‌توان پرونده‌های ذخیره شده را نیز ویرایش کرد.

قیمت و تاریخ ورود سورفیس به بازار هنوز اعلام نشده است. مایکروسافت با معرفی این کامپیوتر لوحی وارد بازی‌ای می‌شود که تاکنون تنها اپل بازیگر آن بود. یعنی طراحی و ساخت سخت‌افزار کامپیوتر لوحی به همراه سیستم عامل ویژه آن.

کارشناسان احتمال می‌دهند که شرکت گوگل نیز پس از خریداری شرکت موتورولا، تبدیل به سومین شرکتی شود که شانس خود را در ارائه کامپیوتر لوحی مبتنی بر سیستم عامل خود (اندروئید) امتحان می‌کند.

انتظار می‌رود در دو سال آینده تقاضا برای کامپیوترهای لوحی سه برابر شود و در پایان سال آینده میلادی، فروش سالانه آن‌ها از ۱۸۰ میلیون دستگاه فراتر برود.

بی بی سی، ۱۹ جون ۲۰۱۲

مترجم‌های انسانی در وب

یک شرکت نوپای کامپیوتری به نام دولینگو در ازای آموزش زبان به انسان‌ها از آن‌ها برای ترجمه محتویات وب کمک می‌گیرد. این شرکت توسط لوئیس فون آن، دانشیار دانشگاه کارنگی ملون، تأسیس شده است. دولینگو به علاقه‌مندان فراگیری زبان،

آموزش‌های ابتدایی را ارائه می‌کند و سپس جملاتی را برای ترجمه در اختیارشان می‌گذارد. یکی یکی و از ساده به پیچیده. پدید آورندگان محتوا بر روی وب نیز برای ترجمه مطالبشان می‌توانند از طریق دولینگو از این مترجمان رایگان استفاده کنند. این خدمت فعلاً رایگان است.

زبان‌هایی که در دولینگو آموزش داده می‌شود در حال حاضر فقط انگلیسی، اسپانیایی، فرانسه و آلمانی است. دولینگو از همه متن‌های مجاز برای درس‌هایش استفاده می‌کند و کاربران می‌توانند به بهترین ترجمه‌ها امتیاز بدهند.

به گفته فون آن، کاربران زبانی را فرا می‌گیرند و در همان زمان به ترجمه محتویات وب نیز کمک می‌کنند. این یک نوع آموزش تجربی است.

در نقطه مقابل، سرویس ترجمه گوگل کاملاً ماشینی است و با وجودی که بهبود زیادی یافته اما هنوز از کیفیت مطلوب به دور است. گوگل برای آموزش ماشین‌هایش، حجم زیادی از متن‌های ترجمه شده، از جمله انتشارات سازمان ملل را وارد موتور ترجمه‌اش کرده است.

الگوی مورد استفاده دکتر فون آن، الگوی تازه‌ای است که به «جمع سپاری» یا «گروه سپاری» (Crowdsourcing) شهرت یافته و برخلاف برون سپاری که کار به فرد یا نهاد خاصی واگذار می‌شود در اینجا گروه زیاد و نامشخصی از کاربران اینترنت برای انجام کار مشارکت می‌کنند.

دکتر فون آن، ایده ترجمه را از آنجا به دست آورد که متوجه شد بسیاری از دوستان و خویشاوندان سرزمین مادریش در گواتمالا که تسلطی به زبان انگلیسی ندارند، محتویات بسیار کمتری را در وب در دسترس دارند.

تاکنون روزنامه نیویورک تایمز دولینگو را برای ترجمه محتویات دیجیتالی‌ش به سایر زبان‌ها مورد آزمایش قرار داده اما هنوز به طور رسمی به این سرویس روی نیاورده است.

دکتر فون آن به عنوان نخستین پروژه ترجمه

به ویکی پدیا فکر می‌کند. ویکی پدیا در حال حاضر نزدیک به ۴ میلیون مطلب به زبان انگلیسی دارد که این میزان بسیار بیشتر از سایر زبان‌هاست. پس از آن، آلمانی با ۱/۴ میلیون مطلب، فرانسه با ۱/۳ میلیون مطلب، هلندی با یک میلیون مطلب و اسپانیایی با ۹۰۰/۰۰۰ مطلب قرار دارند.

سخنگوی ویکی پدیا اعلام داشته که از ترجمه مطالب ویکی پدیا استقبال می‌کند اما مطالب ترجمه شده باید پیش از آن که در دائره‌المعارف ویکی پدیا قرار گیرند از نظر صحت و دقت ترجمه مورد بررسی قرار گیرند.

در دولینگو یک متن بزرگ توسط کامپیوتر به بخش‌های آسان و سخت تقسیم می‌شود و به فراگیران زبان در سطح‌های مختلف برای ترجمه داده می‌شود. سپس قسمت‌های ترجمه شده دوباره توسط کامپیوتر به همدیگر متصل می‌شوند.

دولینگو با ۳/۳ میلیون دلار سرمایه که از طریق سرمایه‌گذاران خطرپذیر تأمین شده شکل گرفته است.

نیویورک تایمز، ۱۹ جون ۲۰۱۲

رقابت اپل با گوگل در زمینه نقشه‌سیار

هفته گذشته شرکت اپل اعلام کرد که از پاییز امسال دیگر تلفن‌های این شرکت از نقشه‌های گوگل استفاده نخواهند کرد و به جای آن از سرویس نقشه خود اپل که به صورت توکار در سیستم عامل جدید این شرکت یعنی iOS 6 قرار داده شده استفاده خواهند کرد.

برای بسیاری از مردم، تلفن‌های همراه به صورت ابزار مهمی برای سیر دنیا در آمده‌اند و نقشه‌های سیار در کانون این سرویس قرار دارند. در حال حاضر، گوگل در دنیای نقشه‌های سیار بیشترین سهم را دارد و نقشه‌هایش بر روی تلفن‌های همراه آیفون اپل و البته هر تلفن همراهی که از سیستم عامل اندروئید گوگل استفاده می‌کند قرار دارد.

پیچیده‌تر شدن نرم‌افزارهای سرقت از بانک

دو تا از خطرناک‌ترین نرم‌افزارهای سرقت پول از حساب‌های بانکی، ارتقاء یافته و اکنون قادر به انتقال پول به صورت خودکار و بدون نظارت رخنه‌گر شده‌اند.

به گفته پژوهشگران شرکت ژاپنی ترندمیکرو که در زمینه امنیت سیستم‌های کامپیوتری فعالیت می‌کند، آخرین نسخه‌های برنامه‌های Zeus و SpyEye توانسته‌اند در یکبار ۱۳۰۰۰ یورو از یک حساب برداشت کنند.

پژوهشگران این شرکت، حملات جدید را در دهه‌ها مؤسسه مالی در آلمان، انگلستان و ایتالیا رصد کرده‌اند. این مسأله از آن نظر مشکل‌ساز است که بانک‌های اروپایی معمولاً فناوری‌های دفاعی پیشرفته‌تری نسبت به بانک‌های آمریکایی دارند و بنابراین بانک‌های آمریکایی ساده‌تر مورد حمله قرار خواهند گرفت.

با آن که Zeus و SpyEye توسط گروه‌های متفاوتی نوشته شده‌اند اما در این قابلیت مشترکند که می‌توانند بر روی کامپیوترهایی که وبگاه‌های آلوده یا صفحات تحت کنترل رخنه‌گران را بازدید می‌کنند نصب شوند. این دو برنامه در بازار زیرزمینی رخنه‌گران به فروش می‌رسند و سپس توسط خریداران، مناسب‌سازی شده یا با پودمان‌های (ماجول) اضافی بهبود می‌یابند.

این برنامه‌ها از روشی به نام "تزریق در وب" برای تولید حوزه‌های داده‌ای اضافی به هنگام ورود به سیستم بانکی یا سایر وبگاه‌های حساس توسط قربانی استفاده می‌کنند. برای مثال، کاربر به جای آن که حوزه‌های شماره حساب و رمز عبور را به هنگام ورود به سیستم ببیند، درخواستی برای وارد کردن شماره کارت عابر بانکش را نیز دریافت می‌کند. سپس رخنه‌گر از این اطلاعات برای ورود به حساب بانکی قربانی در آینده و انتقال وجه استفاده می‌کند.

معمولاً انتقال پول زمانبر است و رخنه‌گر مبلغی را انتقال می‌دهد که زیاد جلب توجه نکند.

بنابر اعلام گوگل، این یک راه حل کامل برای مشکلات امنیتی در اینترنت نیست زیرا تهدیدهای امنیتی دائماً در حال تکامل هستند اما فناوری‌ها و پردازش‌های ما نیز روز به روز بهتر می‌شوند.

یاهو نیوز، ۱۸ جون ۲۰۱۲

رکورد جدیدی برای شکستن رمزنگاری جفت پایه

پژوهشگران آزمایشگاه‌های فوجیتسو، مرکز ملی فناوری اطلاعات و ارتباطات ژاپن و دانشگاه کیوشو با شکستن یک کلید ۹۲۳ بیتی به کار رفته در سیستم رمزنگاری جفت پایه (pairing-based)، رکورد جهانی جدیدی برپا کردند.

پژوهشگرانی که در این پروژه کار می‌کردند موفق شدند کلید رمزبندی ۹۲۳ بیتی را در ۱۴۸/۲ روز با استفاده از ۲۱ کامپیوتر و مجموعاً ۲۵۲ پردازنده بشکنند. پیش از این تخمین زده می‌شد که شکستن رمزنگاری جفت پایه با این طول به چند صد هزار سال زمان نیاز داشته باشد.

پژوهشگران این پروژه توضیح داده‌اند که این مسأله را با استفاده از چند فناوری جدید، شامل روش‌های بهینه‌سازی، یک الگوریتم جدید جستجوی دوبعدی و برنامه‌نویسی موازی حل کرده‌اند.

رمزنگاری جفت پایه برای کاربردهایی به کار می‌رود که برای آن‌ها رمزنگاری سنتی کلید عمومی نامناسب باشد. این روش رمزنگاری از سال ۲۰۰۰ توجه رمزنگاران را به خود جلب کرده است اما امنیت آن هنوز به طور کامل مورد مطالعه قرار نگرفته است.

پژوهشگران امیدوارند تلاش‌هایشان به سازمان‌های استانداردسازی و دولت‌ها در تعیین نقشی که رمزنگاری جفت پایه در توسعه استانداردهای رمزنگاری نسل بعد ایفاء خواهد کرد کمک کند.

آی تی ورلد، ۱۹ جون ۲۰۱۲

این اقدام اپل پیش بینی شده بود زیرا این شرکت از مدتی قبل به خریداری چند شرکتی که در این زمینه فعالیت می‌کنند پرداخته بود. از جمله حق دسترسی به داده‌های مربوط به نام و شکل خیابان‌ها در بیش از ۱۰۰ کشور جهان را از شرکت هلندی تام‌تام به دست آورده بود.

اکنون پرسشی که پیش می‌آید این است که آیا اپل خواهد توانست سرویسی به خوبی گوگل یا بهتر از آن ارائه کند یا خیر؟ اپل تاکنون در این زمینه هیچ محصولی نداشته است.

هم اکنون ۲۴۰ برنامه کاربردی مختلف برای تلفن‌های همراه بر پایه سرویس نقشه نوشته شده است که ۷۳ درصد بیشتر از سال قبل و ۲۴۳ درصد بیشتر از سال ۲۰۰۹ است.

گوگل تاکنون اعلام نکرده که چه هزینه‌ای برای سرویس نقشه خود پرداخته اما ناظران حدس می‌زنند که سالانه بین ۵۰۰ میلیون تا یک میلیارد دلار، یعنی یک پنجم کل بودجه تحقیق و توسعه این شرکت صرف آن می‌شود.

یاهو نیوز، ۱۷ جون ۲۰۱۲

کشف روزانه ده هزار وبگاه مخرب توسط گوگل

بنابر اعلام شرکت گوگل، یکی از وظایف جویشر این شرکت، دورنگاه داشتن کاربران اینترنت از بدافزارهاست. این شرکت با حفاظت‌های تعبیه شده در مرورگرهای کروم، فایرفاکس و سافاری، به ۶۰۰ میلیون کاربر خود در مورد وبگاه‌های خطرناک هشدار می‌دهد.

گوگل روزانه بیش از ۹۵۰۰ وبگاه جدید کشف می‌کند که برای توزیع بدافزارها طراحی شده‌اند و یا عامل آن‌ها قرار گرفته‌اند. در حدود ۱۲ تا ۱۴ میلیون جستجو و ۳۰۰ هزار بارگیری در روز، هشدار امنیتی دریافت می‌کنند.

کنند. هنگامی که کارسازهای تحت اختیار نویسندگان Flame اخیراً دستور انهدام کد را فرستادند، سیمانتک بلافاصله آن را تشخیص داد.

فرمان از بین بردن خود، پرونده‌ای به نام "browse32.ocx" بود. هنگامی که این پرونده اجرا می‌شد، به‌طور خودکار تک بیت‌های Flame را مکان‌یابی می‌کرد، آن‌ها را پاک می‌کرد و داده‌های تصادفی را بر روی کد اصلی می‌نوشت.

به نظر می‌رسد که دستور انهدام پس از آن که ویروس Flame شناسایی و به‌طور گسترده گزارش شد صادر شده باشد.

یاهونیوز، ۸ جون ۲۰۱۲

خرید شرکت کوئست توسط دل

شرکت دل، دومین تولیدکننده بزرگ رایانه‌های شخصی در آمریکا پس از هیولت پاکارد، شرکت نرم‌افزاری کوئست را به مبلغ ۲/۴ میلیارد دلار خریداری کرد. شرکت کوئست ۱۳۰۰ مهندس دارد و به طراحی نرم‌افزار برای بنگاه‌های دولتی و شرکت‌های خصوصی می‌پردازد. دل امیدوار است با خریداری کوئست بتواند کارسازها (سرویسگر)، تجهیزات حافظه و شبکه، و خدمات رایانشی بیشتری به فروش رساند. همه این‌ها این روزها سودآورتر از فروش رایانه‌های شخصی هستند زیرا افراد بیشتر بر تلفن‌های هوشمند و رایانه‌های لوحی تکیه می‌کنند تا رایانه‌های شخصی.

خریداری کوئست، اندازه بخش نرم‌افزار شرکت دل را دو برابر می‌کند. درآمد شرکت کوئست در سال گذشته ۸۵۷ میلیون دلار بوده است.

شرکت کوئست بزرگترین خرید دل پس از خریداری شرکت مشاوره پروت سیستمز در سال ۲۰۰۹ به مبلغ ۳/۹ میلیارد دلار بوده است.

آسوشیتدپرس، ۳ جولای ۲۰۱۲

پیش‌بینی می‌شود که گوگل بر تلاش‌هایش در زمینه ارائه دستگاه‌های خود بیفزاید. از جمله، پیش‌بینی می‌شود تا پایان این ماه، گوگل کامپیوتر لوحی خود را به نام نکسوس به نمایش گذارد. برنامه‌ریزی برای تولید تلفن هوشمند Siri که رقیب آیفون خواهد بود نیز در این شرکت شتاب گرفته است.

وال استریت جورنال، ۱۶ جون ۲۰۱۲

بدترین ویروس کامپیوتری کد خود را از بین می‌برد

هر کسی که بدافزار Flame (شعله) را برای آلوده کردن کامپیوترها در خاورمیانه طراحی کرده ظاهراً نمی‌خواهد کد آن مورد تحلیل قرار گیرد.

شرکت سیمانتک که در زمینه امنیت سیستم‌های کامپیوتری فعالیت می‌کند و ویروس Flame را مورد مطالعه قرار داده است اعلام کرد که کارسازهای (سرویس‌گر) کنترل‌کننده ویروس، دستور از بین بردن خود را ارسال کرده‌اند. این دستور برای از بین بردن هر گونه ردی از کد در کامپیوترهای آلوده طراحی شده است.

نویسندگان Flame این بدافزار را از طریق کارسازهای فرمان و کنترل (C & C) اجرا می‌کنند. برخی از این کارسازها تحت کنترل سیمانتک قرار گرفته است اما بقیه همچنان در دست نویسندگان Flame قرار دارند.

سیمانتک از کامپیوترهای "طعمه" استفاده کرده است. کامپیوترهایی که عمداً با بدافزار آلوده می‌شوند تا پژوهشگران بتوانند به مطالعه رفتار آن بپردازند. درست مثل ویروس‌های زیست‌شناختی که در آزمایشگاه‌ها مورد تحلیل قرار می‌گیرند.

کامپیوترهای آلوده شده به بدافزار Flame، به‌طور مستمر با کارسازهای C & C تماس می‌گیرند تا دستورهای جدید را دریافت

اخیراً برخی بانک‌ها برای افزودن امنیت سیستم از رمزعبورهای یک‌بار مصرف استفاده می‌کنند و آن را از طریق پیامک به تلفن همراه مشتری، هنگام درخواست ورود به سیستم، می‌فرستند.

فروش برنامه‌های Zeus و SpyEye عمدتاً در روسیه، اکراین و رومانی صورت می‌گیرد و نرم‌افزار جدید بین ۳۰۰ تا ۴۰۰۰ دلار به فروش می‌رسد. البته خریدار باید هزینه‌های مناسب‌سازی را جداگانه پرداخت کند.

رویترز، ۱۸ جون ۲۰۱۲

خریداری شرکت یامر توسط مایکروسافت

مایکروسافت شرکت نرم‌افزاری یامر را به قیمت یک میلیارد دلار خریداری کرد. شرکت یامر در سال ۲۰۰۸ تأسیس شد و در زمینه خدمات شبکه‌های اجتماعی سازمانی فعالیت می‌کرد. این شرکت موفق شده بود ۱۴۲ میلیون دلار سرمایه از طریق سرمایه‌گذاران خطرپذیر جذب کند.

بیش از ۸۰ درصد از ۵۰۰ شرکت لیست فورچون (Fortune ۵۰۰) مشتری محصول شرکت یامر هستند.

وال استریت جورنال، ۱۵ جون ۲۰۱۲

تلفن هوشمند گوگل در راه است

رقابت بین اپل و گوگل هرگز تا بدین حد شدید نبوده است. با وجودی که گوگل سهم بیشتری از بازار سیستم عامل تلفن‌های هوشمند را در اختیار دارد اما تلفن آیفون اپل، محصول بسیار سودآورتری از سیستم عامل اندروید گوگل است. سیستم عامل اندروید گوگل متن باز است و با وجودی که همراه با جویشگر گوگل و برخی خدمات دیگر عرضه می‌شود اما گوگل هنوز راهی برای کسب درآمد از آن بجز از راه تبلیغات نیافته است.

سما: سامانه‌ای به کمک رایانه، نظارت شده، تکاملی و دورگه، برای خدمات حسابداری مدیریت

سید ابراهیم ابطاحی

عضو هیأت علمی دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف

پست الکترونیکی: abtahi@sharif.edu

مقدمه

برخی از کارکردهای بانک‌ها و سازمان‌های امین^۱، شامل خدماتی برای مدیریت حسابداری در اجرای پروژه‌های بزرگ است. تامین تضمینی درخواست مالی مجریان پروژه در طی اجرا و مدیریت جریان نقدینگی و چرخه‌های واسط بین پیمانکاران و کارفرمایان که بتواند منجر به تضمین تحقق محدودیت‌های پروژه در منابعی نظیر هزینه و زمان شود، استفاده از این خدمات را در پروژه‌های بزرگ سرشار از قوت و فرصت می‌نماید.

حسابداران مدیریت شریک‌های راهبردی همزمان ساز چرخه‌های پروژه‌ای و گردش‌های مالی، ارزش آفرینانی بانگهای آینده‌نگر هستند که با اعتناء به ارزش زمانی پول و بهره‌گیری از منابع راکد در بازه‌های زمانی، فرصت‌های تجاری خلق کرده و از ثمرات آن برای زمان‌های احتمالی مواجهه پروژه با تنگنای مالی و اجرایی و اداری، استفاده می‌کنند.

حوزه‌های فعالیت حسابداری مدیریت حداقل در شش زمینه تخمین نیازهای سرمایه در گردش^۲، تامین مالی سرمایه در گردش^۳، مدیریت سرمایه در گردش^۴، مدیریت حساب‌های دریافتنی^۵، مدیریت موجودی^۶ و مدیریت دارائی‌های جاری^۷ تعریف می‌شود و می‌تواند در

چکیده

سامانه پیشنهادی موضوع این مقاله (سما)*، محیطی به کمک رایانه است که می‌تواند استفاده بیشینه‌ای از رایانه برای ارائه خدمات حسابداری مدیریت** یا حسابداری مدیریتی*** بنماید. سما با اخذ داده‌های خام و تولید اطلاعات مورد نیاز حسابداری مدیریت به شکل برگه‌های روزآمد درخواستی و داشبورد وضعیت نمای مدیریتی، تصمیمات و توصیه‌های مدیریتی را برپایه تجارب کارشناسی قابل اتخاذ کرده و ثمربخشی نتایج حاصله را قابل سنجش می‌نماید. این سامانه با حضور فعال خبره یا خبرگان تصمیم‌گیر و توصیه‌ساز حسابداری مدیریت و به کمک اطلاع‌سازان مجری، زمینه فعالیت‌های حسابداری مدیریت را برای خبرگان فراهم می‌سازد و با کمک گروه دوم در طی اجرای پروژه درخواست‌های اطلاعاتی خبره یا خبرگان را محقق نموده و تصمیمات و توصیه‌های آن‌ها را در قبال توفیقاتشان از لحاظ ثمربخشی می‌سنجد. داده‌های سما از دو سامانه همکار حسابداری مالی و مدیریت پروژه اخذ و پس از انجام فرآیندهای تعریف و تدوین شده، براساس نیازهای درخواستی خبره در قالب برگه‌های هوشمند و یا شاخص‌های توفیق در داشبورد مدیریتی، آن‌ها را به خبره یا خبرگان حسابداری مدیریت عرضه می‌نماید.

کلید واژه‌ها

حسابداری مدیریت، سازمان‌های امین، داشبورد مدیریتی، سنجش ثمربخشی، شاخص‌های وضعیت نما، برگه‌های هوشمند اطلاعاتی.

* SEMA: Supervised Evolutionary computer aided Management

Accounting Services system

** Management Accounting

***Managerial Accounting

1-Trust Banks & Organizations

2-Estimation of Working Capital Requirement - Financing Working Capital

3- Financing Working Capital

4 - Management of Working Capital

5-Management of Account Receivables

6-Management of Inventory

7- Management of Current Assets

این فرآورده‌هاست تا تصمیمات حسابداری مدیریت در راستای تحقق پروژه قرار گیرد.

با توجه به ماهیت تصمیمات و توصیه‌ها که به خبرگی و پیشینه دانشی کارشناس خبره حسابداری مدیریت وابسته است در گونه‌هایی از این نوع از سامانه‌ها از جمله سما به ماهیت نظارت شده به معنی حضور فیزیکی حذف‌ناپذیر کارشناس خبره انسانی و تکاملی بودن به معنی اعمال تصمیمات و توصیه‌ها با سعی و خطا در محیط واقعی باید اذعان داشت و بر مبنای آن چارچوب سامانه را یک راه حل "به کمک رایانه"^۸ برآورد نمود و این کمک را در قالب یک داشبورد وضعیت نمای پروژه برای مدیران خبره حسابداری مدیریت کامل کرد.

سرعت مورد انتظار در تبادلات داده‌ای در عین وابستگی اطلاعاتی به دو سامانه مستقل حسابداری مالی و مدیریت پروژه ایجاد می‌کند از سهل‌ترین فناوری‌های رایج انتقال داده استفاده شود تا عملاً اجرای مستمر آن ممکن باشد. به این دلیل برگه‌های اطلاعاتی به عنوان رسانه تبادل داده‌ای انتخاب گردیده که دو گونه داده خام^۹ و اطلاعات هوشمند^{۱۰} آن مراجع اطلاعاتی این سامانه باشند. گونه‌های خام بر مبنای ملاحظات نیازسنجی شده آغازین و درخواست حین پروژه (براساس درخواست خبره حسابداری مدیریت) مستمراً تولید می‌شوند و گونه هوشمند این برگه‌ها براساس درخواست‌های مدیر خبره حسابداری مدیریت جهت استفاده مستمر او توسط راهبران سامانه تولید و بر روی سامانه درج می‌گردد. با ملاحظات فوق این سامانه بر یک وبگاه اینترنتی نصب و راه اندازی می‌شود که همه ذینفعان شامل راهبران، تولیدکنندگان اطلاعات، خبرگان حسابداری مدیریت و مدیران ارشد و اجرائی پروژه به آن دسترسی داشته و آن را به هنگام کرده و بتوانند از خدمات آن بهره گیرند.

مولفه‌های سما

سما، سامانه‌ای تکاملی و نظارت شده از گونه "سامانه‌های خبره نیمه خودکار"^{۱۱} است. اخذ و درج نیازهای داده‌ای و محاسباتی، فرآیند پردازش اطلاعات به کمک رایانه و رده‌بندی تصمیم‌ها و توصیه‌های خبرگی از خدمات این سامانه در حوزه حسابداری مدیریت است. پنج مولفه اصلی این سامانه شامل ذینفعان آن، داشبورد وضعیت نمای مدیریتی، برگه‌های نیازهای داده‌ای خام، برگه‌های هوشمند اطلاعات پرورده و صورت سنجش ثمربخشی شده تصمیم‌ها و توصیه‌های خبرگی مدیر خبره حسابداری مدیریت به عنوان ذینفع اصلی سما است.

ذینفع اصلی سما، مدیر خبره حسابداری مدیریت به عنوان عاملی انسانی است که با بیان و درج نیازهای داده‌ای، امکان تولید برگه‌های داده خام مورد نیاز از سامانه‌های حسابداری مالی و مدیریت پروژه را فراهم می‌سازد و با بیان نیازهای محاسباتی، با ارسال این برگه‌ها

راستای خلق فرصت‌های تجاری به فعالیت‌های جزئی‌تری نیز تقسیم شود.

نیازهای سرمایه در گردش از طریق بررسی چرخه عملیات هم‌زمان پروژه، با بررسی چرخه تبدیل نقدینگی آغاز می‌شود و با سنجش ثمربخشی تأثیر زمان بر نیازمندی‌های گردش سرمایه می‌تواند به اتخاذ رویکردهائی برای برآورد میزان سرمایه در گردش منجر شود. برای تأمین این برآورد، فعالیت تأمین مالی سرمایه در گردش کارساز است که با بررسی انواع گوناگون راه‌های تأمین مالی آغاز و با محاسبه نرخ‌های مؤثر سود وام بانکی و هزینه تخفیفات مالی پیگیری می‌شود. بررسی انواع منابع مالی با (یا بدون) وثیقه، کوتاه و میان مدت برای تأمین مالی حساب‌های دریافتنی می‌تواند فعالیت بعدی در این زمینه باشد. با تعیین فرآیند خرید و فروش بدهی‌ها و تأمین مالی تهیه موجودی‌ها می‌توان در این گام به تعیین راهبردهای تأمین مالی دست یافت. فعالیت مدیریت سرمایه در گردش به عنوان فعالیتی دیگر در حوزه حسابداری مدیریت به بررسی سرمایه در گردش می‌پردازد و با شناسائی عوامل تأثیرگذار و ویژگی‌های آن در حوزه‌های مختلف کاری پایان می‌گیرد.

ساماندهی حساب‌های دریافتنی از طریق مدیریت آن‌ها از دیگر فعالیت‌های کلیدی حسابداری مدیریت است که به طراحی یک خط‌مشی در این زمینه و ارزیابی آن منجر می‌شود و از طریق تجزیه و تحلیل دوره‌ای بدهکاران به بررسی امکانات خرید و فروش بدهی‌ها منجر می‌گردد.

در حوزه منابع، حسابداری مدیریت به مدیریت موجودی‌ها و دارایی‌های جاری اعتناء دارد که در اولی با تهیه پایه‌های موجودی و هزینه‌های آن و تعیین سطوح مختلف موجودی‌ها، زمان‌های سفارش موجودی و میزان سفارش‌ها را معین کرده و با تجزیه و تحلیل تخفیف‌ها و کیفیت سفارش‌ها، اولویت‌ها را به مجریان توصیه می‌کند. فعالیت مدیریت دارائی‌های جاری با تعیین مولفه‌های مدیریت نقدینگی آغاز شده و به تهیه بودجه نقدینگی ختم می‌شود. بررسی چرخه‌های تبدیل نقدینگی در این فعالیت می‌تواند به انتخاب مدل‌های مدیریت نقدینگی منجر شود که مثلاً با مدیریت غلطان نقدینگی به یافتن سطح بهینه‌ای از نقدینگی لازم منجر می‌گردد [۱]، [۲].

چارچوب سما

حسابداری مدیریت در چهار حوزه کلان: مدیریت دارائی‌ها، مدیریت حساب‌های دریافتنی، مدیریت موجودی‌ها، مدیریت تأمین مالی و برآورد نیازهای سرمایه در گردش، نیازمند اخذ مستمر اطلاعات از سامانه حسابداری مالی است. در عین حال با توجه به ماهیت تصمیمات و توصیه‌های کارشناسان خبره حسابداری مدیریت در جهت راهبردهای پروژه مستلزم اخذ اطلاعاتی از سامانه مدیریت پروژه در رابطه با فعالیت‌های کلان، منابع مصروفه، فرآورده‌های حاصله و کیفیت کمینه مورد انتظار

8- Computer Aided

9-Dumb

10-Smart

11-Semi-automated Expert Systems

برگه‌های هوشمند اطلاعات پرورده شده بر مبنای برگه‌های داده‌ای خام و نیازهای محاسباتی توصیفی-متنی کاربر خبره حسابداری مدیریت، از طریق راهبران و اطلاع‌سازان سما، تولید شده و بر پایگاه اینترنتی حامل سما برای استفاده مدیر خبره حسابداری مدیریت درج می‌شوند. سنجش ثمربخشی تصمیم‌ها و توصیه‌های مدیریتی، مولفه پایانی سما است که از طریق تصمیم‌ها و توصیه‌های انسانی اخذ شده توسط مدیر خبره حسابداری مدیریت با درج در سما شکل می‌گیرند و سما آن‌ها را از طریق تحلیل نتایج به کمک راهبران و اطلاع‌سازان رده‌بندی کرده و صورت اولویت‌دار توفیقاتشان را بر پایگاه اینترنتی سما درج می‌کند.

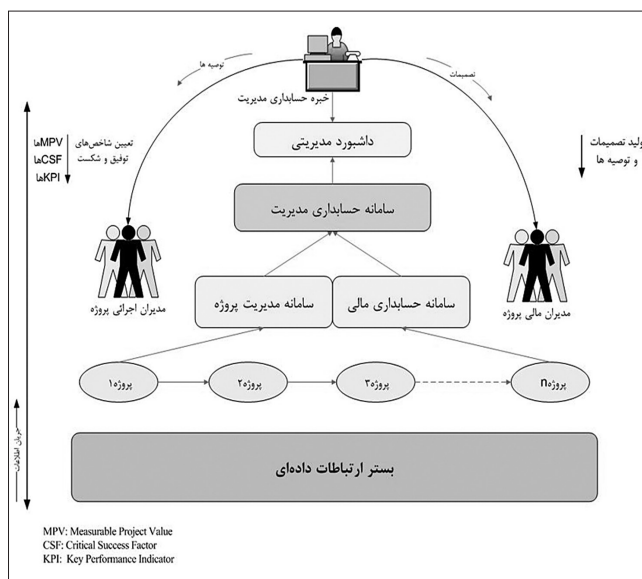
شمای کلان سما

شکل یک، جریان اطلاعاتی را نشان می‌دهد که بر بستر ارتباطات داده‌ای از پروژه‌ها به سوی سامانه‌های حسابداری مدیریت، بر حسب نیاز، واکنشی شده و نتایج تحلیلی آن بر اساس تصمیم‌ها و توصیه‌های انسانی خبره حسابداری مدیریت بر داشبورد مدیریتی درج می‌شود. این داشبورد نمایانگر برآورد شاخص‌های توفیق و یا شکست مرحله‌ای، ادواری یا نهایی پروژه است. نتایج و علائم مندرج بر داشبورد مدیریتی هم برای مدیران اجرایی و هم مدیران مالی پروژه قابل دسترسی و استفاده خواهد بود [۴]. سما شامل چهار طبقه خدمت پایه: نیازهای اطلاعاتی سامانه‌ای (حسابداری مدیریت، مدیریت منابع و شاخص‌ها)، داشبورد وضعیت نمای مدیریتی، خدمات برگه‌های خام داده‌ای و برگه‌های هوشمند اطلاعاتی و آرشیو رتبه‌بندی شده تصمیم‌ها و توصیه‌های خبرگی است.

معماری و گردش کار سما

معماری سما که سامانه‌ای دورگه^{۱۲} متشکل از عوامل انسانی و نرم‌افزاری است، از یک چرخه درخواست داده و تولید اطلاع، اخذ داده از سامانه‌های مربوط، آماده‌سازی برگه‌های هوشمند اطلاعاتی توسط راهبران، درج بر پایگاه اینترنتی سما، استفاده توسط خبره حسابداری مدیریت، اخذ تصمیم یا توصیه مدیریتی، اعمال و سنجش ثمربخشی آن‌ها تشکیل شده است. این چرخه در عین حال همراه با یک فرآیند موازی و مستمر و لحظه‌ای وضعیت نمای شاخص‌ها بر داشبورد وضعیت نمای مدیریتی برخوردار است (شکل دو). نیازسنجی داده‌ای و محاسباتی این چرخه در نوبت اول توسط کارشناس سیستم صورت می‌گیرد و در دفعات بعد به شکل اظهار و درخواستی، کاربر خبره حسابداری مدیریت آن را برای راهبران سما ارسال می‌کند که با درج برگه‌های داده‌ای خام، درخواست نوسازی داده‌ها و اطلاعات و تولید برگه‌های اطلاعاتی هوشمند پردازش شده، اطلاعات لازم برای سما و داشبورد وضعیت نمای مدیریتی فراهم می‌نمایند.

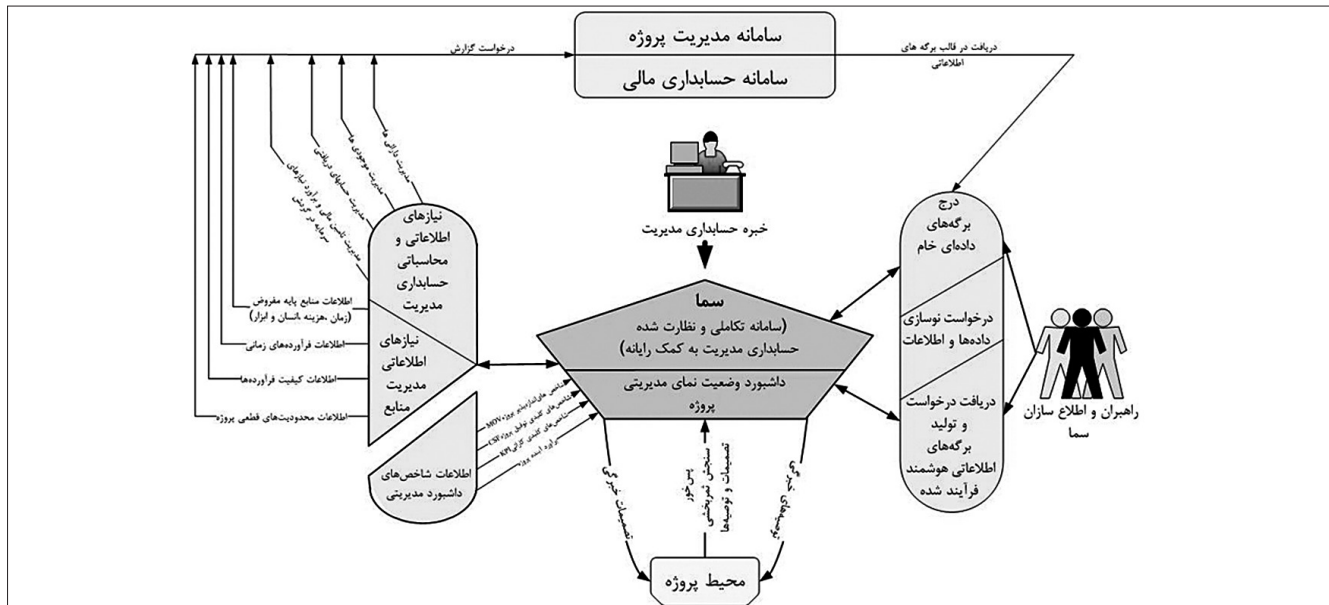
اخذ داده‌ها از سامانه‌های حسابداری مالی و مدیریت پروژه در قالب



شکل ۱- شمای کلان سما

برای گروه راهبران سما که در طول مدت اجرای پروژه، بر روی پایگاه اینترنتی حامل سما، پاسخگوی فرد خبره هستند امکان تولید برگه‌های اطلاعاتی هوشمند مورد نیاز را فراهم می‌سازد که آن‌ها را در اسرع وقت بر پایگاه اینترنتی حاوی سما برای استفاده فرد خبره درج نمایند. داشبورد وضعیت نمای مدیریتی سما که بر اساس اطلاعات شاخص‌های تولید شده توسط مدیر خبره حسابداری مدیریت، خدمات تصویری خود را عرضه می‌نماید نمایشگر شاخص‌های کلیدی کارائی، در ازای عوامل کلیدی توفیق و در چارچوب ارزش‌های اندازه‌پذیر پروژه هستند که براساس پیش‌فهم (پارادایم) جدید داشبورد مدیریتی تعریف خواهند شد. در پیش‌فهم کهن داشبورد مدیریتی تاکید بر سودآوری با کمینه کردن هزینه‌ها بود که ابزار اصلی در آن سرمایه‌گذاری تلقی می‌شد، اما در پیش‌فهم جدید، بازنمائی سودآوری در قبال کیفیت و رضایت عوامل انسانی مرتبط در اولویت قرار دارد که به تبع آن دارائی‌ها، بدهی و تعهدات، شاخص‌های اصلی قابل بازنمائی در سطح داشبورد مدیریتی محسوب می‌گردند [۳].

برگه‌های خام نیازهای داده‌ای، سومین مولفه مهم سما هستند که به تفکیک حوزه‌های مرتبط حسابداری مدیریت یعنی مدیریت دارائی‌ها، موجودی‌ها، حساب‌های دریافتنی، تامین و برآورد مالی نیازهای سرمایه در گردش، حاوی برگه‌هایی بر پایگاه اینترنتی سما هستند که تعریف هر برگه با نیازسنجی از کاربر خبره حسابداری مدیریت در ابتدا انجام شده و بر روی سما تعریف می‌شود و با اخذ اطلاعات از سامانه‌های حسابداری مالی و مدیریت پروژه تکمیل می‌گردد. در گام‌های بعد، کاربر خبره حسابداری مدیریت اقلام برگه‌های داده‌ای خام جدید مورد نیاز خود را تعیین کرده و با تعریف فرآیندهای پردازشی مورد نیاز آن‌ها، امکان تولید برگه‌های هوشمند اطلاعاتی را برای راهبران و اطلاع‌سازان سما فراهم می‌سازد تا با تولید این برگه‌های اطلاعاتی هوشمند تولیدی، آن‌ها را بر پایگاه اینترنتی سما درج کنند.



شکل ۲- معماری و گردش کار سما

نصب و راهاندازی سما، راهبری و نگهداری سما شامل آموزش منابع انسانی کاربر و راهبر سما، آغاز دوره نگهداری و نوسازی داده ها و اطلاعات و خدمات سما برحسب درخواست های داده ای و اطلاعاتی کاربر اصلی خبره حسابداری مدیریت.

نتیجه گیری

توفیق سما، مستلزم امکان اخذ اطلاعات از دو سامانه حسابداری مالی و مدیریت پروژه است که به همین منظور توصیه می گردد تولیدکنندگان سما در مرحله انتخاب نرم افزارهای مذکور جهت بررسی امکان اخذ اطلاعات از درون این سامانه ها، مورد مشورت واقع شوند. سادگی معماری سما، حسن آن است و فرآورده های داده ای و اطلاعاتی و وضعیت نمای آن (که باید تلاش شود تا در نهایت سادگی، سهولت و سرعت دسترسی بیشتری را ممکن سازند) امکانات گسترده ای را برای کاربران فراهم می آورند. این سامانه به لحاظ ماهیت نیمه خودکار خود علی الظاهر واجد نقطه ضعفی است که قوت تکامل پذیری نظارت شده آن به جای تهدید، فرصت های بسیاری برای کاربران آن فراهم می سازد. شرط توفیق سما، حفظ نگاهی کمینه گرا در دسترسی به اطلاعات و داده ها و فرآیندهای محاسباتی لازم و تکیه بر گستردگی حوزه ها به لحاظ پوشش بیشتر خدمات حسابداری مدیریت می تواند باشد.

مراجع و منابع

- [1] Ronald Hilton(2008), "Managerial Accounting: Creating Value in a Dynamic Business Environment", McGraw-Hill.
- [2] Jerry J. Weygandt(2009), "Managerial Accounting: Tools for Business Decision Making", Wiley.
- [3] Inge Hanschke(2010), "Strategic IT Management : A Toolkit for Enterprise Architecture Management", Springer.
- [4] J.T. Marchewka(2006), "Information Technology Project Management: Providing Measurable Organizational Value", Second Edition, Wiley.

برگه های اطلاعاتی صورت می گیرد و در همین قالب پس از محاسبه به عنوان برگه اطلاعاتی هوشمند بر پایگاه اینترنتی سما بازنمایی می شود. سما که با حداقل خدمات و اطلاعات پس از نیازسنجی اولیه در اسرع وقت بر پایگاهی اینترنتی جهت استفاده کاربر خبره حسابداری مدیریت برپا خواهد شد، طی دوران کاربری تکاملی و نظارتی، گسترش یافته و به بلوغ و غنای لازم خواهد رسید.

چرخه برپائی سما

گام های برپائی سما، با سامانه های اطلاعاتی متعارف اندکی متفاوت است زیرا در آن، تولید، برپائی، راهبری و نگهداری پیوندی ناگسستنی دارند. نیازسنجی این پروژه فعالیت تخصصی ویژه ای را می طلبد و طراحی و پیاده سازی آن ملاحظات خاصی دارد که در عین حال به علت تکاملی و نظارتی بودن با راهبری و نگهداری پایگاه حامل اطلاعات و خدمات سما هم پیوند دارد. چرخه برپائی سما که در زنجیره برپائی، راهبری و نگهداری تکرار شونده در طول عمر پروژه مورد مطالعه، از گام های مکملی نظیر مشاوره و اعلام نظر در مورد تناسب سامانه های حسابداری مالی و مدیریت پروژه به علت حفظ هم سازی با نیازهای سما هم تشکیل شده است، به شرح زیر می باشد:

نیازسنجی داده ای، اطلاعاتی و فرآیندی حسابداری مدیریت، نیازسنجی خدماتی حسابداری مدیریت، مشاوره و اعلام نظر برای انتخاب سامانه های حسابداری مالی و مدیریت پروژه، نمونه سازی مولفه های مناسب اخذ داده از دو سامانه فوق، استخراج نیازهای شاخص های وضعیت نمای داشبورد مدیریتی، طراحی برگه های داده ای، اطلاعاتی و خدماتی سما، طراحی و تولید پایگاه اینترنتی حاوی سما و برگه های آن، پیاده سازی کدهای برنامه ای و نمونه سازی و آزمون یک چرخه کمینه برپائی تمامی خدمات سما،

بیانیه منابع آموزشی آزاد (OER¹)

یونسکو - پاریس: ۲۰ تا ۲۲ ژوئن ۲۰۱۲

ترجمه: دکتر فریده مشایخ
پست الکترونیکی: f.mashayekh@pedagogy.ir

غیررسمی) از دیدگاه یادگیری مداوم و حمایت از انسجام اجتماعی، برابری جنسی و نیازهای ویژه آموزشی، بهبود هزینه اثربخشی و کیفیت پیامدهای یاددهی - یادگیری از طریق به کارگیری OER.

۲- تسهیل توانمندسازی محیط‌های یادگیری برای استفاده از فناوری‌های اطلاعاتی و ارتباطی (ICT)، کاهش فاصله دیجیتالی با استقرار زیربنای مناسب به ویژه از طریق افزایش پهنای باند برقراری اتصال و گسترش فناوری موبایل. ارتقاء سواد رسانه‌ای و اطلاعاتی^۳ (MIL) و تشویق توسعه به کارگیری OER در قالب تولیدات دیجیتالی باز و استاندارد.

۳- تقویت توسعه راهبردها و سیاست‌ها در حوزه OER. ارتقاء توسعه سیاست‌های ویژه برای تولید و استفاده از OER در قالب راهبردهایی برای ارتقاء آموزش.

۴- ارتقاء درک استفاده از چارچوب‌های «گواهینامه خلاقیت مشترک^۴ (CCL)». حمایت از باز به کارگیری، بازبینی، باز ترکیب و باز توزیع منابع آموزشی در جهان از طریق گواهینامه خلاقیت مشترک (CCL) همزمان با دفاع از حق تألیف و حقوق معنوی.

3- media and information literacy

4- Common Creative License Lhttp:// CreativeCommons.org)

مقدمه مترجم

گذار به جامعه اطلاعاتی و تحقق «جامعه دانش - بنیان» در هزاره سوم میلادی مستلزم بازاندیشی در بسیاری از مفاهیم از جمله مفهوم عدالت در یادگیری - به معنای فراهم ساختن امکان دسترسی برابر - در مقیاس ملی و جهانی است. در این ارتباط، در حوزه آموزش، طی دهه گذشته، سازمان یونسکو به منظور پیشگیری از «فاصله دیجیتالی» و فراهم ساختن زیرساخت‌های لازم برای دسترسی جهانی به اطلاعات و دانش، به زمینه‌سازی توأم با آگاه‌سازی، و تدوین کنوانسیون‌های بین‌المللی پرداخته است. با شکل‌گیری جنبش «منابع آموزش آزاد» در کنگره‌ای که طی سه روز در خرداد ماه ۲۰۱۲ در پاریس / یونسکو با حضور نمایندگان کشورهای عضو از جمله ایران تشکیل شد، بیانیه‌ای حاوی ۱۰ توصیه^۱، برای دولت‌ها تدوین و انتشار یافته است که در ذیل ترجمه آن برای خوانندگان گزارش کامپیوتر ارائه می‌گردد:

*

*

۱- ترویج آگاهی و استفاده از منابع آموزشی آزاد. ارتقاء و گسترش دسترسی به آموزش در تمام سطوح (آموزش‌های رسمی و

۱- برای آشنایی با مفهوم OER رجوع شود به گزارش کامپیوتر، سال سی‌وسوم، خرداد و تیر ۱۳۹۰، شماره ۱۹۷، ص ۷۲
2- 2012 World Open Educational Resources (OER) Congress, UNESCO, Paris June 20-22, 2012

با حمایت از تنوع فرهنگ‌ها، به دانش و حقوق بومی احترام بگذارند.

۸- تشویق پژوهش در حوزه OER. ترویج تحقیق و توسعه (R&D) در به‌کارگیری، ارزیابی و سازگارسازی محتوا با بسترهای جدید در OER و فرصت‌ها و چالش‌هایی که ایجاد می‌شود و تأثیر آن‌ها بر کیفیت و هزینه اثربخشی و یاددهی - یادگیری، به منظور تقویت شواهد برای سرمایه‌گذاری بخش عمومی در OER.

۹- تسهیل دسترسی، جستجو، پردازش و سهیم شدن در OER. تضمین توسعه ابزار کاربرپسند برای یافتن و پردازش OER متناسب با نیازهای ویژه کاربران. گزینش استانداردهای مناسب باز (open) برای تضمین هم‌کنش‌پذیری و تسهیل استفاده از OER در رسانه‌های گوناگون.

۱۰- تشویق و حمایت از گواهینامه محتوا آزاد برای مواد آموزشی باز با هزینه عمومی. دولت‌ها و مقامات با صلاحیت می‌توانند سودمندی و سودآوری مواد آموزشی باز را برای شهروندان تضمین نمایند.

۵- حمایت از توانمندسازی برای توسعه پایدار «مواد یادگیری با کیفیت»^۵. حمایت از نهادهای کارآموزی و تشویق معلمان و دیگر کارکنان آموزشی و پژوهشی در تولید و سهیم شدن در به‌کارگیری منابع آموزشی باکیفیت، با توجه به نیازهای محلی و تنوع یادگیرندگان. ارتقاء و تضمین کیفیت مرور منابع آموزشی آزاد. تشویق توسعه سازوکارهایی برای سنجش و گواهی پیامدهای یادگیری از طریق به‌کارگیری OER.

۶- ترویج ارتباطات راهبردی برای OER. استفاده از فناوری برای ایجاد فرصت در سهیم شدن در مواردی که با گواهینامه «محتوا آزاد» در زمینه‌های گوناگون رسانه‌ای تولید شده و از طریق همکاری‌های راهبردی پایداری میان آموزش، صنعت، کتابخانه، رسانه و بخش‌های ارتباطاتی را تضمین می‌کند.

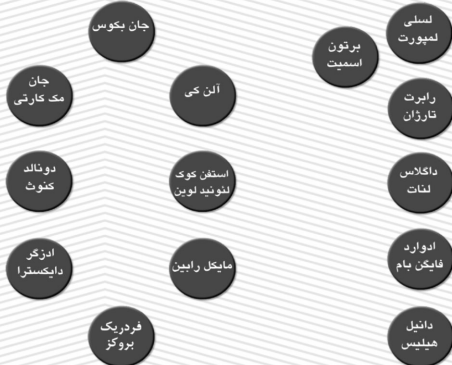
۷- تشویق توسعه و سازگارسازی OER در زبان‌ها و بسترهای فرهنگی گوناگون، حمایت از تولید و استفاده از OER در زبان‌های محلی و تضمین دسترسی. سازمان‌های دولتی لازم است همراه

5- quality learning materials

بزرگان دانش رایانه زندگی و نوآوری‌های پانزده دانشمند علوم رایانه



نویسندگان: دنیس الیوت شاشا
کتی لازر
ترجمه ابراهیم نقیب زاده مشایخ



دبیرخانه شورای عالی اطلاع‌رسانی

قیمت: ۲۵۰۰۰ ریال

شابک ۹۶۴-۸۸۳۶-۱۹-۷ | ISBN 964-8846-19-7

انجمن انفورماتیک ایران

محاسبات کوانتومی: از نظریه تا زبان‌های برنامه‌سازی

مصطفی برمشوری

کارشناس ارشد علوم کامپیوتر، دانشگاه تهران

پست الکترونیکی: mostafa.barmshory@p-simorgh.com

محمد هادی منصوری

دانشجوی کارشناسی ارشد علوم کامپیوتر، دانشگاه تهران

پست الکترونیکی: h.mansouri@p-simorgh.com

چکیده

امروزه رایانه‌ها بر اساس فناوری‌های متفاوتی ایجاد شده و به بازار عرضه می‌شوند. پردازشگرهای این رایانه‌ها بر اساس هر فناوری که باشد از نظر محاسباتی با یکدیگر تفاوت چندانی ندارند و در عمل همگی ماشین تورینگ با نوار متناهی را به نحوی پیاده‌سازی می‌کنند. واضح است که مبانی مورد استفاده در زبان‌های برنامه‌سازی برای این ماشین‌ها با یکدیگر تفاوت چندانی ندارد و تفاوت‌های موجود تنها در مجموعه دستورالعمل‌هایی است که به کار می‌برند. اما محاسبات و مکانیک کوانتومی گونه‌ای جدید از پردازشگرها را معرفی کرده است که نه تنها در فناوری بلکه در مبانی نیز با پردازشگرهای کلاسیک متفاوتند. بدیهی است که این پردازشگر جدید نیاز به زبان‌های برنامه‌سازی جدیدی دارد. در این مقاله زبان‌های برنامه‌سازی کوانتومی و معماری مورد حمایت آن‌ها بررسی شده و یک دسته‌بندی مناسب برای این زبان‌ها ارائه شده است.

مقدمه

رایانه‌های امروزی همگی بر اساس ماشین محاسباتی تورینگ طراحی و با استفاده از ترانزیستورها و قطعات الکترونیکی دیگر پیاده‌سازی شده‌اند. در فناوری‌های اخیر، سعی در کاهش اندازه ترانزیستورهای استفاده شده در پردازشگرهای جدید به عمل آمده، اما این کاهش

اندازه تا کجا می‌تواند ادامه یابد؟ اگر ترانزیستورها خیلی کوچک شوند، آنگاه تأثیرات عجیب دنیای کوانتومی، کارایی آن‌ها را تحت تأثیر قرار خواهد داد. حتی اگر الگوریتم‌های موفق کوانتومی هم موجود نباشد، فناوری مورد استفاده محدود است و به ناچار باید به سوی مکانیک کوانتومی، به عنوان بستری برای پیاده‌سازی پردازشگرها رفت. تأثیر مکانیک کوانتومی بر روی محاسبات و اطلاعات بیش از آن چیزی است که بتوان تصور کرد. کاربرد مکانیک کوانتومی در محاسبات و اطلاعات منجر به پیدایش دو شاخه جدید به نام محاسبات و اطلاعات کوانتومی شده است.

اطلاعات کوانتومی شاخه‌ای از علم است که در آن به بررسی روش‌های انتقال اطلاعات، با استفاده از اصول مکانیک کوانتومی پرداخته می‌شود. مهمترین اصل مکانیک کوانتومی که باعث ایجاد توانایی‌های جدید در اطلاعات کوانتومی شده است، درهم تافتگی^۱ است. درهم تافتگی حالتی است که در آن نمی‌توان حالات ذرات یک سیستم را مستقل از هم تصور کرد. روش‌هایی مثل کدگذاری فوق چگال^۲ و

1- Entanglement

2- Superdense coding

بار مقداردهی تابع انجام می‌شود، در حالی که در محاسبات کلاسیک راهی جز مقداردهی به ازای تمام حالت‌های ورودی ممکن کشف نشده است. با کار دویچ بود که برتری محاسبات کوانتومی نسبت به محاسبات کلاسیک اثبات شد. پس از آن، دویچ به همراه جوزا^۹ این روش را توسعه دادند و توانستند با همین سرعت این خصوصیت را برای توابع با برد بیشتر هم به کار ببرند [۴].

در سال ۱۹۸۲ ریچارد فاینمن^{۱۰} ایده رایانه کوانتومی را مطرح کرد [۹]. رایانه‌ای که از اثرات مکانیک کوانتومی بهره‌برداری کند. تا مدتی این ایده فقط از نظر تئوری جالب بود، تا این که پیتر شور^{۱۱} در سال ۱۹۹۴ الگوریتمی کوانتومی، برای تجزیه عددها به عوامل اول اختراع کرد [۳]. روش شور قادر است در زمانی از مرتبه چند جمله‌ای، یک عدد را به عوامل اول تجزیه کند. در رایانه‌های کلاسیک اعتقاد بر این است که تجزیه به عوامل اول کاری سخت است و تجزیه اعداد بزرگ روی یک رایانه کلاسیک - حتی یک ابر رایانه - می‌تواند چندین سال یا حتی چندین قرن طول بکشد، در حالی که روی یک رایانه کوانتومی، می‌تواند در چند دقیقه انجام شود. این موضوع زمانی اهمیت بیشتری می‌یابد که بدانیم بسیاری از الگوریتم‌های رمزنگاری امروزی از جمله RSA، که از پرکاربردترین الگوریتم‌های رمز کلاسیک است، بر اساس سختی محاسبه عوامل اول کار می‌کند. در صورت ساخت رایانه کوانتومی، این گونه رمزنگاری‌ها در مدت زمانی کوتاه شکسته خواهند شد. چند سال بعد از شور، گزارش‌هایی دال بر پیاده‌سازی الگوریتم شور توسط گروه‌های متفاوت اعلام شد. برای مثال می‌توان به کار شرکت آی‌بی‌ام اشاره کرد. آی‌بی‌ام با استفاده از یک محاسبه‌گر کوانتومی NMR، توانست عددهای چهار بیتی را به عوامل اول تجزیه کند. پژوهشگران شرکت آی‌بی‌ام توانستند عدد ۱۵ را به عوامل اول آن یعنی ۳ و ۵ تجزیه کنند. شاید خنده‌دار باشد اما در دنیای علم این اتفاق تحولی بسیار بزرگ است [۱۰].

کمی بعد از شور یکی دیگر از الگوریتم‌های موفق کوانتومی طراحی شد. گروور الگوریتم جستجویی را معرفی کرد که می‌توانست از میان داده‌های نامرتب در زمانی از مرتبه $O(\sqrt{n})$ داده مورد نظر را پیدا کند [۵]. در برخی از موارد این سرعت بسیار جالب توجه است اما در برخی دیگر نه. سریع‌ترین الگوریتم جستجوی کلاسیک روش جستجوی دودویی است [۱۱]، که از میان داده‌های مرتب با $O(\log n)$ می‌تواند داده مورد نظر را بیابد. مشخص است که در این حالت روش کوانتومی ناکارآمد است. اما زمانی که داده‌ها نامرتب است روش کوانتومی کارایی خوبی دارد. شور بعدها نشان داد که در پایگاه‌های داده بزرگ، که امکان پرسش در مورد چند داده ممکن است، می‌توان تنها با یک پرسش مکان داده مورد نظر را تعیین کرد [۱۲]. این کار در نوع خود بسیار کارآمدتر از روش‌های کلاسیک است.

دورفرستی کوانتومی^۲ بر همین اساس شکل گرفته‌اند. ابرحالت‌ها^۴ نیز در مکانیک کوانتومی امکانات جدیدی را در توزیع کلید کوانتومی به وجود آورده‌اند که تاکنون راهی برای درهم شکستن آن یافت نشده است [۱].

نظریه محاسبات کوانتومی نیز شاخه‌ای از علم است که در آن به بررسی الگوریتم‌های کوانتومی و سخت‌افزارهای کوانتومی مورد نیاز برای پیاده‌سازی عملی آن‌ها پرداخته می‌شود. پیچیدگی‌های محاسباتی بر اساس مدل‌های متفاوتی مطرح می‌شوند، اما مهمترین اصل مورد استفاده در این الگوریتم‌ها دست یافتن به کارآیی بهتر نسبت به معادل‌های کلاسیک است. نشان داده شده است که الگوریتم‌ها و سیستم‌هایی که از دو اصل مکانیک کوانتومی درهم‌تافتگی و ابرحالت‌ها استفاده نکنند الگوریتم‌هایی معمولی به حساب می‌آیند [۲]. بنابراین می‌توان حدس زد که محاسبات کوانتومی تنها بر اساس استفاده از همین دو اصل، نسبت به محاسبات کلاسیک کارآمدتر است. الگوریتم‌های تجزیه به عوامل اول [۳]، تعیین توابع متوازن و ثابت [۴]، جستجوی گروور [۵] و تبدیل فوری [۳] تنها بخشی از الگوریتم‌های موفق هستند که بسیار کارآمدتر از روش‌های کلاسیک مطرح شده می‌باشند.

چارلز بنت^۵ فیزیکدان شرکت آی‌بی‌ام در سال ۱۹۷۳ نشان داد که می‌توان محاسبات را به صورت برگشت‌پذیر^۶ انجام داد [۶]. این دستاورد را نمی‌توان سرآغاز پردازش‌های کوانتومی دانست اما از آن جهت که تحول سیستم‌ها در مکانیک کوانتومی به صورت تبدیل‌های برگشت‌پذیر است در حالی که پردازش‌های کلاسیک بر اساس تبدیلات غیربرگشت‌پذیرند، می‌توان این کار را به عنوان پیش‌نیازی برای پردازش کوانتومی دانست.

آغاز رسمی محاسبات و اطلاعات کوانتومی با کار پل بنیوف^۷ در سال ۱۹۸۰ بود [۷]. یکی از اصول به کار رفته در مکانیک کوانتومی تحول سیستم‌ها بر اساس تبدیلات یکانی است. بنیوف در کار خود نشان داد که تبدیلات یکانی معادل با همان دروازه‌های منطقی مورد استفاده در محاسبات است و بنابراین می‌توان محاسبات را با استفاده از این تبدیلات انجام داد.

در سال ۱۹۸۵، چند سال بعد از مقاله بنیوف، دیوید دویچ^۸ یک قدم بسیار مهم در زمینه محاسبات کوانتومی برداشت [۸]. وی در مقاله خود نشان داد که چگونه می‌توان درهم‌تافتگی کوانتومی را برای رسیدن به محاسبات سریع‌تر مورد استفاده قرار داد. دویچ بر همین اساس روش معروف خود را در تعیین توابع ثابت و متوازن معرفی کرد و نشان داد که محاسبات کوانتومی بسیار کارآمدتر از محاسبات کلاسیک است. در روش دویچ، تعیین توابع ثابت و متوازن تنها با یک

3- Quantum Teleportation

4- Superposition

5- Charles Bennett

6- reversible

7- Paul Benioff

8- David Deutsch

9- Richard Jozsa

10-Richard Feynman

11-Peter Shor

به سادگی وابستگی هر دوی آن‌ها را به علم فیزیک نشان داد. دستاوردهای اطلاعات و رایانش به ناچار باید در یک بستر فیزیکی پیاده‌سازی شوند، از همین رو دور از تصور نیست که تغییرهای بنیادی در علم فیزیک منجر به پیدایش تغییر اساسی در علم‌های رایانش و اطلاعات شود.

در ۱۹۲۴ لوئیز دوبروی دوگانگی موجی- ذره ای را مطرح کرد؛ ذرات می‌توانند خاصیت‌های موجی از خود به نمایش بگذارند و برعکس. با معرفی روش دوبروی و پس از آن در ۱۹۲۵ با توسعه مکانیک ماتریسی توسط ورنر هایزنبرگ، ماکس بورن و پاسکال یوردان و اختراع مکانیک موجی و یک معادله غیر نسبیتی- به عنوان حالت عمومی نظریه دوبروی- توسط اروین شرودینگر، مکانیک کوانتومی متولد شد. شرودینگر بعداً نشان داد این دو روش معادل هستند. پیشرفت در این حوزه همچنان ادامه داشت. هایزنبرگ اصل عدم قطعیت را در ۱۹۲۷ فرمول‌بندی کرد. در ابتدای ۱۹۲۷، پل دیراک اقدام به یکی کردن مکانیک کوانتومی با نسبیت خاص کرد و معادله دیراک برای الکترون، مطرح شد. معادله دیراک توصیفی نسبیتی از تابع موج یک الکترون ارائه می‌دهد که شرودینگر در به دست آوردن آن ناموفق بود. روش دیراک در نمادگذاری مکانیک کوانتومی بسیار موفق بود تا جایی که امروزه در تمام منابع مکانیک کوانتومی از این روش استفاده می‌شود. در همان دوره جان فون نویمان، پایه ریاضی قدرتمندی را برای مکانیک کوانتومی، با عنوان نظریه عملگرهای خطی روی فضاهای هیلبرت فرمول‌بندی کرد.

انیشتین به همراه دو تن از همکارانش، بوریس پودولسکی و ناتان روزن، در ۱۹۳۵ مقاله‌ای معروف- که به مقاله EPR مشهور است - را با عنوان «آیا توصیف مکانیک کوانتومی از واقعیت فیزیکی کامل است؟» ارائه کردند [۱۳]. آنها در این مقاله به اثبات کامل نبودن نظریه مکانیک کوانتومی پرداخته بودند.

بیشتر فیزیکدانان دلایل مقاله EPR را دلایل قانع کننده‌ای نمی‌دانستند. تقریباً یک سال بعد نیلز بوهر در مقاله‌ای با همان عنوان مقاله EPR، به آن پاسخ داد [۱۴] که این پاسخ بیشتر بحثی فلسفی بود. بعدها در سال ۱۹۵۱ دیوید بوهم [۱۵] و [۱۶]، تناقض EPR را به صورتی دیگر بیان کرد.

محتوای مقاله EPR و پاسخ بوهر بیشتر به عنوان بحث بین دیدگاه‌های فلسفی متفاوت بود. اما در سال ۱۹۶۵ جان استوارت بل برای اولین بار بحث‌های کیفی EPR را به صورت کمی بیان کرد و این مساله را فرمول‌بندی کرد [۱۷] و [۱۸].

بل با توجه به فرضیات EPR (یعنی موضوعیت و واقعیت مستقل از مشاهده)، روابطی به دست آورد - معروف به نامساوی بل - که در شرایطی خاص، این روابط با پیش‌بینی‌های مکانیک کوانتومی اختلاف داشت و در نتیجه با آزمایش می‌توان بین این دو نظریه قضاوت کرد. اصول آزمایش بل اگرچه ساده به نظر می‌رسید، اما در عمل، انجام آزمایش بسیار پیچیده است. با این حال آزمایش‌های

با کشف الگوریتم شور توجه بسیاری از پژوهشگران در سراسر دنیا به ایده رایانه‌های کوانتومی جلب شد و در حال حاضر در بسیاری از مراکز تحقیقاتی و با صرف هزینه‌های زیاد این موضوع در حال پیگیری است. اما قدرت یک رایانه کوانتومی در چیست؟ در رایانه‌های کوانتومی بیت به معنای کلاسیک وجود ندارد. در این رایانه‌ها از کیوبیت استفاده می‌شود. یک بیت کلاسیک تنها می‌تواند در یکی از ۲ حالت ممکن - صفر یا یک - باشد و به طور کلی n بیت کلاسیک می‌تواند در یکی از 2^n حالت ممکن باشد، در حالی که یک کیوبیت می‌تواند صفر، یک، هم صفر-هم یک و نه صفر-نه یک (به عبارتی صفر، یک، هر دو، هیچکدام) باشد. در واقع می‌توان گفت قدرت محاسباتی رایانه‌های کوانتومی در موازی‌سازی کوانتومی فوق‌العاده آن‌ها است. شکی نیست که دیگر نمی‌توان امید چندانی به تغییرات اساسی در محاسبات کلاسیک داشت اما با این حال هنوز هم تعدادی زیادی از مسائل باقی مانده است که برای علوم مختلف بسیار اساسی است. گرچه روش‌های جدید محاسبات، امیدهای جدیدی را به وجود آورده‌اند اما هنوز هم توانایی ساخت ماشین‌های محاسبه‌گری که بتوانند با این روش‌های جدید بهینه‌تر از سیستم‌های رقمی کنونی محاسبات را انجام دهند در پرده ابهام است.

مدل محاسباتی مدارهای کوانتومی نمی‌تواند یک مدل محاسباتی برای یک پردازشگر کوانتومی جهانی باشد. پردازشگر کوانتومی جهانی، پردازشگری کوانتومی است که می‌تواند هر پردازش کوانتومی را پیاده‌سازی کند. ایجاد یک الگوی مناسب برای بیان الگوریتم‌های کوانتومی بسیار ضروری است. روش‌های پردازش و همچنین الگوریتم‌ها عموماً بر اساس مدل‌های محاسباتی خاص توسعه پیدا می‌کنند. بیان الگوریتم‌ها بر اساس مدل‌های محاسباتی، توانایی پیاده‌سازی سخت‌افزاری را راحت‌تر می‌کند. از این رو پیش از ایجاد روش‌هایی برای بیان الگوریتم‌های کوانتومی لازم است که یک مدل محاسباتی مناسب تعیین شود.

در ادامه پس از بررسی مبانی مکانیک و محاسبات کوانتومی، معماری QRAM تشریح شده و پس از آن زبان‌های برنامه‌سازی کوانتومی که بر اساس این معماری شکل گرفته‌اند، معرفی و دسته‌بندی خواهند شد.

۲- محاسبات کوانتومی

کارهای ارزنده شانون منجر به پیدایش علم اطلاعات شد در حالی که دوازده سال پیش از آن نیز علم رایانش با کارهای ارزنده تورینگ معرفی شده بود. علم رایانش و اطلاعات به عنوان پدیده‌هایی بسیار نو، منجر به پیدایش تحول‌های بزرگی در تمام زمینه‌های علمی و اقتصادی شدند. اما با نگاهی سطحی به این دو دستاورد مهم علمی، یعنی علم رایانش و اطلاعات، می‌توان

۲-۲ حافظه کوانتومی

یک بیت کوانتومی یا کیوبیت، معادل با حالت کوانتومی یک سیستم فیزیکی است که از دو حالت پایه تشکیل شده است. در حالت کلاسیک می‌توان آن را معادل با یک سیستم فیزیکی فرض کرد که از دو حالت متفاوت مانند $\{0,1\}$ تشکیل شده است. در محاسبات کوانتومی نمادگذاری دیراک مورد استفاده قرار می‌گیرد از این رو حالت‌های پایه سیستم کوانتومی مورد نظر به صورت $\{|0\rangle, |1\rangle\}$ نمایش داده می‌شود.

هر یک از حالت‌های پایه معادل با یک حالت کلاسیک درستی و نادرستی است که می‌تواند با استفاده از خصوصیت‌های فیزیکی کوانتومی مانند اسپین هسته و یا حالت فوتون نمایش داده شود. در اینجا جزئیات فیزیکی یک کیوبیت مد نظر نیست اما بر اساس مبانی کوانتومی یک سیستم کوانتومی نه تنها می‌تواند در حالت‌های پایه‌ای خود باشد بلکه می‌تواند در یک ترکیب خطی از حالت‌های پایه نیز قرار داشته باشد. بر این اساس حالت کلی یک کیوبیت به صورت $\alpha|0\rangle + \beta|1\rangle$ نمایش داده می‌شود که در آن داریم:

(۱)

$$|\alpha|^2 + |\beta|^2 = 1$$

البته می‌توان حالت هر کیوبیت را در حالت کلی با استفاده از بردار نیز نمایش داد. حالت یک کیوبیت با استفاده از نمایش برداری به صورت زیر خواهد بود:

(۲)

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \alpha|0\rangle + \beta|1\rangle$$

به بیان دیگر حالت یک کیوبیت معادل با بردارهای به طول یک از فضای برداری هیلبرت است. حالت‌های پایه $\{|0\rangle, |1\rangle\}$ یک حالت خاص از تمام پایه‌های یکا متعامد در فضای هیلبرت یک کیوبیت است که به آن‌ها پایه‌های محاسباتی گفته می‌شود. علاوه بر پایه‌های محاسباتی پایه‌های کاربردی دیگری نیز در محاسبات کوانتومی وجود دارد که برای نمونه می‌توان به پایه‌های قطری اشاره کرد که کاربرد فراوانی در اطلاعات کوانتومی و توزیع کلید کوانتومی دارد. در حالت کلی تمام پایه‌ها را می‌توان بر اساس یکدیگر بیان کرد.

در یک سیستم چند کیوبیتی، حالت کلی به صورت یک بردار در فضای تانسوری کیوبیت‌ها در نظر گرفته می‌شود که معادل با همان ضرب کارتیزین در فضاهای معمولی است. فرض کنید که U و V فضاهای برداری با پایه‌های $\{u_i\}$ و $\{v_j\}$ باشند در این صورت فضای تانسوری $U \otimes V$ یک فضای برداری با پایه‌های $\{u_i \otimes v_j\}$ در نظر گرفته می‌شود. از این رو می‌توان گفت که یک سیستم n کیوبیتی معادل با یک فضای برداری با بعد 2^n است که در آن $\{|0\rangle \otimes \dots \otimes |0\rangle, \dots, |0\rangle \otimes \dots \otimes |1\rangle, \dots, |1\rangle \otimes \dots \otimes |1\rangle\}$ پایه‌های فضا را تشکیل می‌دهد. برای سادگی حالت‌های $|a_1\rangle \otimes \dots \otimes |a_n\rangle$ را به صورت $|a_1, \dots, a_n\rangle$ نمایش می‌دهیم.

بسیاری انجام شده که با نتایج مکانیک کوانتومی سازگارند و با نتایج ناشی از فرضیات EPR ناسازگار. این نشان می‌دهد که حداقل یکی از فرض‌های در نظر گرفته شده در نامساوی بل، درست نیست.

۱-۲ مبانی مکانیک کوانتومی

مکانیک کوانتومی بر چهار اصل بنا شده است. اصول مکانیک کوانتومی در واقع ارتباطی بین دنیای فیزیکی و فرمول‌بندی ریاضی مکانیک کوانتومی است. این اصول بعد از یک فرآیند طولانی آزمون و خطا به وجود آمد. به طور خلاصه این اصول عبارتند از [۱۹]:

اصل فضای حالت؛ به هر سیستم فیزیکی یک فضای برداری مختلط با ضرب داخلی (یعنی همان فضای هیلبرت) مربوط می‌شود که به آن فضای حالت می‌گویند.

اصل تحول؛ تحول یک سیستم کوانتومی، با یک انتقال یکانی توصیف می‌شود. از آنجا که هر تبدیل یکانی، یک تبدیل معکوس‌پذیر است تمام تحول‌های کوانتومی برگشت‌پذیر خواهد بود.

اصل اندازه‌گیری؛ اندازه‌گیری‌های کوانتومی با یک گردایه از عملگرهای اندازه‌گیری توصیف می‌شوند که روی فضای حالت سیستم عمل می‌کنند. نتایج به دست آمده از اندازه‌گیری تصادفی است و بسته به نتیجه به دست آمده از اندازه‌گیری حالت سیستم نیز تغییر خواهد کرد.

اصل فضای حالت سیستم‌های مرکب؛ فضای حالت یک سیستم فیزیکی مرکب، ضرب تانسور فضاهای حالت سیستم‌های فیزیکی تشکیل‌دهنده آن است.

تفسیرهای زیادی برای مکانیک کوانتومی وجود دارد که تلاش دارند تصویر روشنی از مفاهیم مکانیک کوانتومی را ارائه دهند. یکی از بحث برانگیزترین موضوعات مکانیک کوانتومی، اصل سوم یعنی اصل اندازه‌گیری است. در مکانیک کوانتومی ویژگی مستقل از مشاهده وجود ندارد، بلکه ویژگی‌های فیزیکی به عنوان نتایجی از اندازه‌گیری‌های انجام شده روی سیستم به دست می‌آیند. بسیاری از فیزیکدانان با این دید مخالف بودند. مشهورترین آن‌ها انیشتین است. انیشتین همیشه با مبانی نظری مکانیک کوانتومی مشکل داشت. جملات معروف او مثل "خدا تاس بازی نمی‌کند"، و بحث‌هایش با بوهر و سایر نظریه‌پردازان مکانیک کوانتومی، اختلاف نظری او را به خوبی نشان می‌دهد.

امروزه مکانیک کوانتومی، به عنوان چهار چوب ریاضیاتی پایه‌ای در بسیاری از زمینه‌های فیزیک و شیمی به کار می‌رود، از جمله: فیزیک حالت جامد، فیزیک اتمی، فیزیک مولکولی، فیزیک محاسباتی، شیمی محاسباتی، شیمی کوانتومی، شیمی هسته‌ای و فیزیک هسته‌ای. علاوه بر این‌ها، فناوری نه چندان دیرپای رایانه نیز از این تحول در دنیای فیزیک بی‌تأثیر نبوده تا جایی که مکانیک کوانتومی به علوم رایانه نیز راه یافته است.

می‌گذارد. برای نمونه عملگر کوانتومی هادامارد برای دو کیوبیت به صورت زیر تعریف می‌شود:

$$H^{\otimes 2} = H \otimes H \quad (9)$$

که تاثیر این عملگر به روی یک سیستم دو کیوبیتی به صورت زیر خواهد بود:

$$\begin{aligned} |00\rangle &\rightarrow |++\rangle \\ |01\rangle &\rightarrow |+-\rangle \\ |10\rangle &\rightarrow |-+\rangle \\ |11\rangle &\rightarrow |--\rangle \end{aligned} \quad (10)$$

البته حالت کیوبیت‌ها می‌تواند به صورت یک ترکیب خطی (یا به صورت یک ابر حالت) از پایه‌های فضای برداری هیلبرت باشد، در این وضعیت، تاثیر عملگرهای یکانی روی یک حالت کلی کوانتومی به صورت زیر تعریف می‌شود:

$$\begin{aligned} U|\psi\rangle &= U\left(\sum_{i=0}^{2^n-1} \alpha_i |i\rangle\right) \\ &= \sum_{i=0}^{2^n-1} \alpha_i U|i\rangle \end{aligned} \quad (11)$$

برای نمونه عملگر یکانی هادامارد به صورت زیر روی پایه‌های قطری عمل می‌کند.

$$\begin{aligned} H|+\rangle &= H\left(\frac{|0\rangle+|1\rangle}{\sqrt{2}}\right) \\ &= \frac{H|0\rangle+H|1\rangle}{\sqrt{2}} \\ &= |0\rangle \end{aligned} \quad (12)$$

یکی از قابلیت‌های مهم مکانیک و محاسبات کوانتومی، اندازه‌گیری کوانتومی است که به عنوان یکی از اصول مکانیک کوانتومی مطرح است. اگر یک کیوبیت در حالت کوانتومی $\alpha|0\rangle + \beta|1\rangle$ باشد، نتیجه اندازه‌گیری آن با احتمال $|\alpha|^2$ مقدار صفر خواهد بود و حالت کوانتومی بعد از اندازه‌گیری به حالت $|0\rangle$ نشانده خواهد شد. به همین ترتیب نیز با احتمال $|\beta|^2$ مقدار یک خواهد بود و حالت کوانتومی سیستم به حالت $|1\rangle$ نشانده خواهد شد.

هر اندازه‌گیری بر اساس یک پایه تعریف می‌شود. برای نمونه در تعریف بالا از اندازه‌گیری پایه‌های محاسباتی به عنوان پایه اندازه‌گیری استفاده شده است. با استفاده از تبدیل‌های یکانی و اندازه‌گیری در پایه استاندارد می‌توان اندازه‌گیری‌های دیگر را پیاده‌سازی کرد. این روش از نظر پیاده‌سازی سیستم‌های فیزیکی از اهمیت برخوردار است.

۳- QRAM

نخستین گام در ایجاد زبان‌های برنامه‌سازی کوانتومی، زبان شبه کدی بود که توسط نیل^{۱۲} در سال ۱۹۹۶ معرفی شد [۲۰ و ۲۱]. هدف اصلی نیل، ایجاد یک زبان برنامه‌سازی حقیقی کوانتومی نبود بلکه تلاش داشت زبان شبه کدی را برای بیان دقیق الگوریتم‌های کوانتومی معرفی کند و امیدوار

هر n کیوبیت را می‌توان معادل با یک ثابت کوانتومی در نظر گرفت که قادر است حالت یک سیستم کوانتومی با n حالت کوانتومی پایه را نمایش دهد. از این ثابت کوانتومی می‌توان در محاسبات کوانتومی استفاده کرد. حالت یک ثابت کوانتومی با n کیوبیت به صورت زیر نمایش داده می‌شود:

$$\begin{aligned} |\psi\rangle &= \sum_{i_0=0}^1 \dots \sum_{i_n=0}^1 \alpha_{i_0 \dots i_n} |i_0\rangle \otimes \dots |i_n\rangle \\ &= \sum_{j=0}^{2^n-1} \alpha_j |j\rangle \end{aligned} \quad (3)$$

که در آن j نمایش دهنده‌ی معادل با حالت پایه‌ای هر یک از کیوبیت‌های موجود در ثابت کوانتومی است.

۲-۳ عملگرهای کوانتومی

تحول یک سیستم ایزوله کوانتومی را می‌توان با استفاده از یک تبدیل یکانی به صورت کامل نمایش داد. اگر حالت کوانتومی با استفاده از بردارهای ستونی نمایش داده شود، آنگاه یک عملگر یکانی را می‌توان با استفاده از یک ماتریس (U_{ij}) نمایش داد که بر روی میدان اعداد مختلط تعریف می‌شود. این تبدیل یکانی در رابطه زیر صدق می‌کند.

$$U^{-1} = U^* \quad (4)$$

که در آن U^* مزدوج ترانپوز ماتریس U است. از اینرو می‌توان تحول سیستم را معادل با اعمال عملگر U بر روی حالت کوانتومی تعریف کرد که با استفاده از ضرب ماتریسی به صورت زیر قابل بیان است:

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \alpha' \\ \beta' \end{pmatrix} \quad (5)$$

یک عملگر کوانتومی را نیز می‌توان با استفاده از تاثیر آن بر روی پایه‌های محاسباتی و یا هر پایه دیگر توصیف کرد. برای نمونه عملگر کوانتومی هادامارد به صورت زیر بر اساس تاثیر آن به روی پایه‌های کوانتومی تعریف می‌شود:

$$\begin{aligned} |0\rangle &\rightarrow \frac{|0\rangle+|1\rangle}{\sqrt{2}} \\ |1\rangle &\rightarrow \frac{|0\rangle-|1\rangle}{\sqrt{2}} \end{aligned} \quad (6)$$

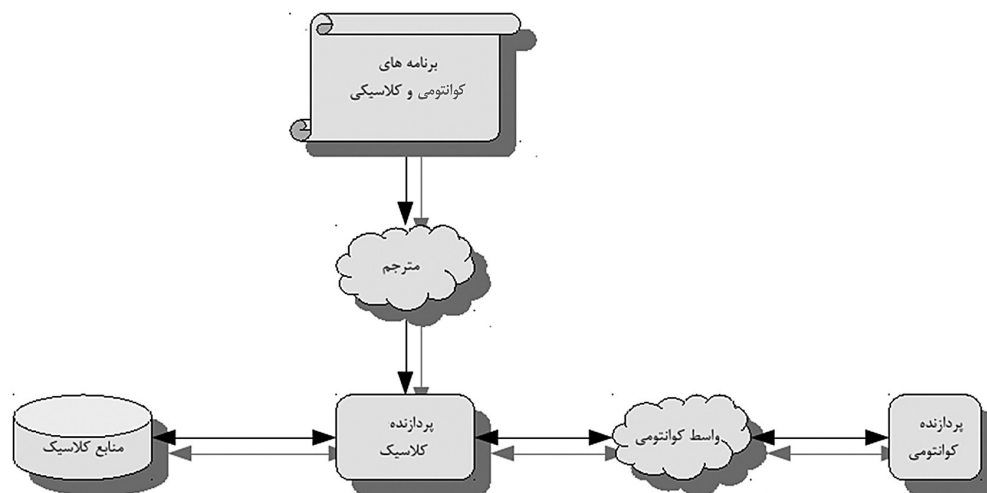
نمایش ماتریسی این عملگر به صورت زیر است:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (7)$$

برای سادگی، تحول‌ها و عملگرهای کوانتومی را با استفاده از یک حرف نمایش می‌دهیم و در رابطه از نوشتن نمایش ماتریسی آن خودداری می‌کنیم. برای نمونه عملگر هادامارد و تاثیر آن به روی حالت کوانتومی به صورت زیر نمایش داده می‌شود:

$$H|0\rangle = |+\rangle \quad (8)$$

معادل با فضای تانسوری برای کیوبیت‌ها، فضای تانسوری عملگرها نیز قابل تعریف است. فضای تانسوری عملگرها، فضای عملگری جدیدی را ایجاد خواهد کرد که روی تعداد کیوبیت بیشتری تاثیر



تصویر ۱: پردازشگرهای کلاسیک به عنوان مدیر کل منابع عمل می کنند. پردازشگرهای کلاسیک با دریافت برنامه های کلاسیک - کوانتومی آن ها را تفکیک می کنند. برنامه های کوانتومی به پردازشگر کوانتومی ارسال می شود در حالی که برنامه های کلاسیک بر روی پردازشگر کلاسیک اجرا می شود.

کوانتومی نه تنها به حالت اولیه مورد نظر ما بارگذاری می شود بلکه بر اساس الگوریتم تعیین شده به گونه ای تحول می یابد که معادل با اجرای الگوریتم کوانتومی مورد نظر است. در نهایت سیستم های کلاسیک با استفاده از حسگرها و دیگر سیستم های اندازه گیری (مانند سیم پیچ های موجود در NMR) داده های تولید شده توسط سیستم کوانتومی را گردآوری می کنند و بعد از اعمال عمل های پردازشی، نتیجه مورد نظر را به دست می آورند. در عمل ساختار مدل محاسباتی QRAM نیز بر اساس همین فرآیند شکل گرفته است.

معماری مدل محاسباتی QRAM پردازشگرهای کلاسیک و کوانتومی را به صورت مدل فرمانده - فرمانبر^{۱۳} با هم آمیخته کرده است. در این معماری پردازشگرهای کلاسیک نقش فرمانده را ایفا می کنند و با ارسال درخواست به سخت افزارهای کوانتومی و دریافت نتیجه، از پردازشگرهای کوانتومی به صورت یک فرمانبر در پردازش ها استفاده می کنند. با استفاده از زبان های برنامه سازی کوانتومی، که به آن ها اشاره کردیم، برنامه هایی ایجاد می شود که شامل دو بخش کوانتومی و کلاسیک هستند. پردازشگر کلاسیک، در فرآیند اجرای برنامه، با رسیدن به دستورات کوانتومی آن ها را برای اجرا به سخت افزارهای کوانتومی ارسال می کند و بعد از دریافت نتیجه به پردازش خود ادامه می دهد. در تصویر شماره ۱ معماری به کار رفته در QRAM نشان داده شده است.

با دریافت دستورات کوانتومی، پردازشگر کوانتومی دستورها را اجرا کرده و در انتها عمل اندازه گیری را انجام می دهد و نتایج به دست آمده را برای پردازشگر کلاسیک ارسال می کند. دریافت دستورها و ارسال نتایج می تواند به صورت خط لوله^{۱۴} و مدام انجام شود. می توان سیستم های پردازشگر کوانتومی را نیز به صورت اشتراکی با چندین پردازشگر کلاسیک به کار برد.

در QRAM، میان ثبات های کوانتومی و کلاسیک تفاوت گذاشته می شود.

بود که این زبان شبیه کد بتواند الگویی برای زبان های برنامه سازی آینده باشد. هر زبان شبه کدی که بر اساس اصول حقیقی موجود در سیستم ها ایجاد شود، به راحتی قابل پیاده سازی است. از این رو نیل، در مقاله اش یک معماری خاص برای سیستم های پردازش کوانتومی در نظر گرفت و بر اساس آن زبان شبه کد کوانتومی خود را معرفی کرد. نیل معماری خود را QRAM نامید. مدل QRAM توسعه یافت تا جایی که امروزه مبنایی برای بسیاری از زبان های برنامه سازی کوانتومی قرار گرفته است.

به صورت مجازی، QRAM همان پردازشگر کلاسیک است با این تفاوت که دسته ای از ثبات های کوانتومی دارد و می تواند روی آن ها عمل های کوانتومی را اجرا کند. منظور از عملگرهای کوانتومی عمل های بارگذاری، اندازه گیری و دسته ای کامل از عمل های تحول کوانتومی است. ساده ترین راه، به صورت مجازی، برای تعریف یک ثبات کوانتومی این است که به روی یک ثبات خاص عملگرهای کوانتومی را تعریف کنیم و همچنین ثبات بتواند حالت های کوانتومی را در خود نگه دارد. داده های کوانتومی موجود در یک ثبات کوانتومی بدون اعمال فرآیند اندازه گیری، قابل دستیابی نیست. در حقیقت ثبات کوانتومی به صورت یک جعبه سیاه است که تا فرآیند اندازه گیری انجام نشود از داده های موجود در آن نمی توان اطلاعی به دست آورد.

قبل از این که به بررسی پیاده سازی یک سیستم پردازشگر کلاسیک که دارای یک یا چند ثبات کوانتومی است بپردازیم اجازه دهید که سیستم های موجود برای پردازش کوانتومی که در آزمایشگاه ها پیاده سازی می شوند را به صورت کوتاه بررسی کنیم. در تمام مدل هایی که تا کنون پیاده سازی شده است فرآیند اجرای الگوریتم های کوانتومی به این قرار است که: ابتدا الگوریتم های کوانتومی با استفاده از پردازشگرهای کلاسیک به دسته ای از تابش ها (الکترومغناطیسی، لیزر و ...) تبدیل می شود، سپس بر اساس واسطه های تعیین شده به یک سیستم کوانتومی (مانند مولکول در NMR و یا یون های در دام افتاده) تابیده می شود. بر اثر تابش ها، سیستم

13- Master-slave

14- Pipe line

که می‌توان از آن به عنوان یکی از جدیدترین زبان‌های برنامه‌سازی کوانتومی یاد کرد. این زبان برنامه‌سازی کوانتومی توسط اوامر معرفی شده است. اوامر در پایان نامه دکتریش اصولی را معرفی کرد که پایه‌ای برای طراحی زبان‌های برنامه‌سازی کوانتومی ساختار یافته است [۲۳ و ۲۴]، مانند زبان‌های برنامه‌سازی ساخت یافته کلاسیک چون C و Pascal. ایجاد زبان برنامه‌سازی کوانتومی QCL به صورت ساخت یافته امکان تعریف عملگرهای کوانتومی جدید را فراهم می‌کند. در زبان برنامه‌سازی کوانتومی QCL داده‌های پایه‌ای کلاسیک همانند داده‌های تعریف شده در زبان‌های برنامه‌سازی کلاسیک، مثل C، ایجاد شده است. علاوه بر آن یک نوع داده‌ای جدید به نام qureg وجود دارد که معادل با یک کیوبیت کوانتومی است. برای ایجاد یک ثابت کوانتومی از همین نوع داده به صورت آرایه استفاده می‌شود. عملگرهای کوانتومی H, T, S و CNOT به صورت از پیش تعریف شده در این زبان برنامه‌سازی وجود دارد (این عملگرها یک دسته عملگر کامل در پردازش‌های کوانتومی را تشکیل می‌دهند).

جانانان گراتیج و تورستن آلتنکریج زبان برنامه‌سازی کوانتومی را ارائه دادند که QML نامیده می‌شود [۲۵]. این زبان یک زبان برنامه‌سازی رویه‌ای است. در بیان ساختار و درستی این زبان برنامه‌سازی، از نظریه رسته‌ها^{۱۶} استفاده شده است. فرض کنید که بتوان یک زبان برنامه‌سازی را با استفاده از یک رسته نمایش داد یا به عبارت دیگر یک رسته معادل با یک زبان برنامه‌سازی کوانتومی ارائه کرد. برای اثبات کامل بودن زبان برنامه‌سازی کوانتومی مورد نظر کافی است نشان دهیم که رسته زبان برنامه‌سازی مورد نظر، هم‌ریخت با رسته پردازش کوانتومی است. این دقیقاً روشی است که در QML استفاده شده است. بر اساس زبان QML یک رسته به نام FQC وجود دارد که می‌توان نشان داد هم‌ریخت با رسته پردازش کوانتومی است. جانانان و تورستن در این کار به این نتیجه رسیده‌اند که اساس زبان‌های برنامه‌سازی کنترل زمان ناهمدوسی است. هاینک زبان برنامه‌سازی امری معرفی کرد [۲۶] و آن را LanQ نامید. این زبان برنامه‌سازی ادغامی میان برنامه‌سازی کلاسیک و برنامه‌سازی کوانتومی است. مدل به کار رفته در این زبان برنامه‌سازی مشابه ساختار پردازش کوانتومی QRAM است. به بیان دیگر پردازش کلاسیک تنها به صورت یک کنترل عمل می‌کند و داده‌ها به صورت کوانتومی معرفی می‌شوند. زبان‌های برنامه‌سازی کوانتومی مانند qGCL و QCL و غیره، بر مبنای ارتباطات کوانتومی طراحی نشده‌اند، لذا در بیان پروتکل‌های چند جانبه کوانتومی مانند BB84 [۱] و یا پروتکل معرفی شده در [۲۷] کاربرد ندارند. از سویی مدیریت پردازش‌ها امری اساسی در پردازش‌های موازی است که در زبان‌های نام برده به این موضوع نیز پرداخته نشده است. البته زبان‌های برنامه‌سازی جبری مانند λ -calculi می‌توانند به روش‌های معادل، پردازش‌های موازی و ارتباط کوانتومی را بیان کنند ولی به یاد داشته باشید که این بیان برای استفاده برنامه‌نویسان مناسب نیست. در زبان برنامه‌سازی LanQ دستوراتی مانند fork، که در مدیریت پردازش‌ها

ثبات‌های کوانتومی توانایی اجرای عمل‌های کوانتومی را دارند. می‌توان یک ثابت کوانتومی را به صورت چندین ثابت کوانتومی کوچک تر فرض کرد و با استفاده از آن‌ها الگوریتم‌های کوانتومی مورد نظر را هم‌زمان، بر روی چندین ثابت کوانتومی فرضی پیاده‌سازی کرد. از سویی می‌توان با استفاده از روندهای نرم‌افزاری با استفاده از یک ثابت کوانتومی امکان برنامه‌نویسی برای چندین ثابت کوانتومی مستقل را فراهم کرد، معادل با آنچه که در زبان‌های برنامه‌سازی کلاسیک روی می‌دهد [۲۲].

چرا مدل پردازشی QRAM یک مدل پردازشی مناسب است؟ نلسن و چانگ نشان داده‌اند پردازش‌هایی که در آن‌ها از احتمالی بودن کوانتومی استفاده نشود با معادل‌های کلاسیک تفاوتی ندارند [۲] و می‌توانند به صورت کلاسیکی انجام شوند. استفاده از پردازشگرهای کلاسیک از دو جهت مفید است: کم‌هزینه‌تر است و طراحی پردازشگر کوانتومی ساده‌تر می‌شود. البته بیان یک مدل محاسباتی به صورتی که کنترل با یک پردازش کلاسیک انجام می‌شود به این معنی نیست که پیاده‌سازی کامل آن با استفاده از پردازشگرهای کوانتومی ممکن نیست. حتی زمانی که پیاده‌سازی یک پردازشگر کوانتومی به صورت کامل، از نظر پیچیدگی و هزینه ممکن باشد باز هم می‌توان در برنامه‌سازی، آن قسمت‌هایی را که می‌تواند به صورت کلاسیک انجام شود به صورت برنامه‌های کلاسیک بیان کرد. بنابراین تصور یک پردازشگر کوانتومی به صورت QRAM می‌تواند مفید باشد.

۴- زبان‌های برنامه‌سازی کوانتومی

تلاش‌های متفاوتی در ایجاد پردازشگرهای کوانتومی با استفاده از اصول مکانیک کوانتومی تا سال ۱۹۸۵ انجام شد، اما هیچ کدام به یک مدل محاسباتی برای پردازش کوانتومی دست نیافتند. دویج در سال ۱۹۸۵ مدل جدیدی از ماشین تورینگ را ارائه کرد که حافظه آن می‌توانست ابرحالت‌های کوانتومی را در خود نگه دارد و نوک^{۱۵} ماشین قادر به اعمال عملگرهای یکانی بر روی خانه‌های حافظه بود [۸]. با این وجود می‌توان گفت که اولین پیشنهاد برای ایجاد یک زبان برنامه‌سازی کوانتومی توسط نیل در سال ۱۹۹۶ معرفی شد [۲۰]. نیل یک شبه کد برای بیان الگوریتم‌های کوانتومی روی یک ماشین کوانتومی با حافظه‌ای با دستیابی تصادفی به نام QRAM، بیان کرد. مدل محاسباتی QRAM، که در گفتارهای بعد به آن خواهیم پرداخت، توسط نیل به صورت دقیق بیان نشده بود. با این حال در مدل QRAM، ماشین دارای ثبات‌های کوانتومی است و قادر به اجرای عمل بارگذاری، عملگرهای یکانی کوانتومی، و اندازه‌گیری کوانتومی روی آن ثبات‌ها است. نیل خود نیز بیان کرده بود که شبه کدش به اندازه کافی قابلیت استفاده به عنوان یک زبان برنامه‌سازی کوانتومی را ندارد، با این وجود کار وی یک قدم بسیار مهم در ایجاد روشی نوین، برای بیان الگوریتم‌ها و استفاده از اصول پایه‌ای مکانیک کوانتومی در پردازش‌ها بود.

یکی از فعال‌ترین زبان‌هایی که پیایی توسعه یافته است، QCL نام دارد

نمونه از بررسی معنایی زبان‌های برنامه‌سازی کوانتومی باشد. واضح است با پیچیده شدن ساختار زبان‌های برنامه‌سازی کوانتومی بررسی معنایی آن‌ها پیچیده و دشوار می‌شود. از این رو برخی زبان‌های برنامه‌سازی کوانتومی به گونه‌ای طراحی شده‌اند که از نظر معنایی بتوان به سادگی آن‌ها را مورد بررسی قرار داد و به همین سبب ساختارهای غیر گویا را به وجود آورده‌اند و در مقابل زبان‌های برنامه‌سازی کوانتومی بسیار گویا به وجود آمده است که از نظر معنایی بررسی نشده‌اند! در دیدگاه ترجمه به بررسی روش‌های ترجمه زبان‌های برنامه‌سازی کوانتومی پرداخته می‌شود و تعیین می‌شود که دشواری ترجمه زبان‌های سطح بالای کوانتومی به زبان‌های سطح پایین و ماشین کوانتومی چیست.

از نظر ساختاری زبان‌های برنامه‌سازی کوانتومی به دسته‌های متفاوتی چون زبان‌های ابتدایی، زبان‌های ساخت یافته، زبان‌های شیء‌گرا و غیره تقسیم می‌شوند. برای نمونه زبان برنامه‌سازی QCL یک زبان برنامه‌سازی ساخت یافته و NDQJava یک زبان برنامه‌سازی شیء‌گرا است. ساختارهای معنایی زبان‌های برنامه‌سازی به روش‌های متفاوتی مورد بررسی قرار می‌گیرد. همان گونه که در مورد زبان برنامه‌سازی QML بیان شد، ساختار معنایی این زبان با استفاده از یک رسته به نام FQC بیان شده و نشان داده شده که این رسته هم‌ریخت با رسته پردازش کوانتومی است. به عبارتی در زبان QML با استفاده از نظریه رسته‌ها، ساختار معنایی مورد بررسی قرار گرفته است. زبان‌هایی که ساختار معنایی آن‌ها به درستی مورد بررسی قرار گرفته‌اند عموماً دارای ساختارهای جبری هستند، که برای استفاده برنامه‌سازان مناسب نیست. از سویی زبان‌هایی که ساختار مناسبی دارند از نظر معنایی به درستی بررسی نشده‌اند!

مهمترین چالشی که در زبان‌های برنامه‌سازی کوانتومی با آن روبرو هستیم، ترجمه زبان‌های برنامه‌سازی به زبان‌های سطح پایین و زبان‌های ماشین است. برای ترجمه زبان‌های سطح پایین و زبان‌هایی که ساختارهای ساده دارند فرآیند ترجمه بسیار ساده است. برای نمونه ترجمه زبان اسمبلی به زبان ماشین تنها به دو بار مرور برنامه در فرآیند ترجمه نیاز دارد و جز فضای داده‌ای مورد نیاز برای خروجی، فضای داده‌ای دیگری نیاز ندارد. این در حالی است که ترجمه زبان‌های برنامه‌سازی سطح بالا به زبان‌های سطح پایین و زبان ماشین بسیار دشوار است. در فرآیند ترجمه زبان‌های کلاسیک به زبان‌های سطح پایین و زبان ماشین، بهینه‌سازی امری بسیار ضروری است. نیاز به فرآیند بهینه‌سازی در ترجمه برنامه‌های کوانتومی بسیار مهم‌تر و حیاتی‌تر است. در پردازنده‌های کوانتومی، علاوه بر عامل‌های محدود کننده‌ای که در پردازنده‌های کلاسیک هم موجود است، مانند زمان و حافظه، زمان نامحدوسی^{۱۷} نیز یک عامل مهم محدود کننده است. زمانی که در پردازنده‌های کلاسیک حافظه پردازنده کم باشد با صرف زمان بیشتر می‌توان به نتیجه درست رسید، اما در مورد یک پردازنده کوانتومی با توجه به زمان همدوسی نمی‌توان این گونه عمل کرد. اگر در یک برنامه کوانتومی زمان مورد نیاز برای اجرا، قبل از فرآیند

مورد استفاده است [۲۸]، پیش‌بینی شده است. علاوه بر این امکاناتی جهت ذخیره کانال کوانتومی و ارسال و دریافت داده‌ها ایجاد شده است که روی هم امکان پیاده‌سازی ارتباطات کوانتومی چندگانه و مدیریت پردازش‌ها و ارتباطات بین پردازشی را در این زبان برنامه‌سازی کوانتومی فراهم کرده است. هاینک بیان کرده است که امکان ترجمه برنامه‌هایی که به زبان QML بیان می‌شوند به برنامه‌هایی که کاملاً بر اساس پردازش‌های کوانتومی باشد وجود دارد، اما به چگونگی آن نپرداخته است.

فمنینگ سونگ و ژیاو خو زبان برنامه‌سازی کوانتومی را بر اساس زبان برنامه‌سازی کلاسیک جاوا ارائه داده‌اند که NDQJava نامیده می‌شود [۲۹]. یکی از اصول مهم توسعه نرم‌افزارهای کلاسیک، گویا بودن برنامه‌های ایجاد شده است، با این وجود زبان‌های برنامه‌سازی کوانتومی مانند QML بسیار پیچیده است، تا جایی که در اولین برخورد نمی‌توان از ساختار و دستورات آن چیزی فهمید. خوانایی برنامه‌های ایجاد شده آنقدر مهم است که به سبب آن تاکنون بسیاری از زبان‌های برنامه‌سازی منسوخ شده‌اند. زبان برنامه‌سازی جاوا [۳۰] یک زبان برنامه‌سازی کلاسیک بسیار پرمفرد در میان برنامه‌نویسان است و تنها تفاوت زبان برنامه‌سازی NDQJava با این زبان، قابلیت برنامه‌نویسی بر اساس اصول مکانیک کوانتومی است. پیاده‌سازی زبان برنامه‌سازی کوانتومی بر اساس زبان برنامه‌سازی جاوا امکان برنامه‌نویسی پیمانه‌ای را، علاوه بر امکان برنامه‌سازی رویه‌ای مانند QCL، فراهم کرده است. یکی از نکات بسیار جالب توجه در این زبان برنامه‌سازی ساختار شیء‌گرای آن است. فمنینگ علاوه بر زبان برنامه‌سازی NDQJava زبان برنامه‌سازی کوانتومی NDOFP را ارائه کرده‌اند که بر اساس زبان برنامه‌نویسی FP است.

۴-۱ دسته‌بندی زبان‌های برنامه‌سازی

زبان‌های برنامه‌سازی کوانتومی که در بالا به آن‌ها اشاره شد تنها بخشی از تمام زبان‌های برنامه‌سازی کوانتومی هستند که در دهه اخیر توسعه داده شده‌اند. بررسی‌های متفاوتی روی زبان‌های برنامه‌سازی کوانتومی انجام شده است [۳۱][۲۲]، که می‌تواند در بررسی درست زبان‌های برنامه‌سازی کوانتومی مفید باشد. زبان‌های برنامه‌سازی کوانتومی را می‌توان بر اساس دیدگاه‌های متفاوتی مورد تحلیل و بررسی قرار داد. مهمترین دیدگاه‌هایی که در این زمینه وجود دارد به صورت فهرست‌وار عبارتند از:

- ساختاری
- معنایی
- ترجمه

دیدگاه ساختاری، به بررسی ساختار برنامه‌سازی و بیان الگوریتم‌های کوانتومی با استفاده از زبان‌های برنامه‌سازی کوانتومی می‌پردازد. برای نمونه بیان شد که زبان برنامه‌سازی کوانتومی QCL یک زبان ساخت یافته است و برنامه‌ها می‌توانند به صورت روال‌های کوانتومی معرفی شوند. دیدگاه معنایی به بررسی درستی و صحت زبان‌های برنامه‌سازی کوانتومی می‌پردازد و روش‌هایی را برای این کار فراهم می‌کند. برای نمونه استفاده از نظریه رسته‌ها در بررسی درستی زبان برنامه‌سازی QML می‌تواند یک

17- Decoherence time

- [6] C. H. Bennett, "Logical reversibility of computation," IBM Journal of Research and Development, vol. 17, pp. 525-532, 1973.
- [7] P. Benioff, "The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines," Journal of Statistical Physics, vol. 22, pp. 563-591, 1980.
- [8] D. Deutsch, "Quantum theory, the Church-Turing principle and the universal quantum computer," Proc. R. Soc. Land. A, vol. 400, pp. 97-117, 1985.
- [9] Feynman, "Simulating physics with computers," International journal of theoretical physics, vol. 21, pp. 467-488, 1982.
- [10] Lieven M. K. Vandersypen, Matthias Steffen, Gregory Breyta, Costantino S. Yannoni, Mark H. Sherwood & Isaac L. Chuang, "Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance," Nature, vol. 414, pp. 883-887, 2001.
- [11] D. E. Knuth, "Optimum binary search trees," Acta Informatica, vol. 1, pp. 14-25, 1971.
- [12] Lov k. Grover, "Quantum Computers Can Search Arbitrarily Large Databases by a Single Query," Phys. Rev. Lett., vol. 79, pp. 4709-4712, 1997.
- [13] A. Einstein, B. Podolsky, and N. Rosen, "Can Quantum Mechanical Description of Physical Reality be Considered Complete?," Phys. Rev., vol. 47, pp. 777-780, 1935.
- [14] N. Bohr, "Can Quantum Mechanical Description of Physical Reality be Considered Complete?," Phys. Rev., vol. 48, pp. 696-702, 1935.
- [15] D. Bohm, "A Suggested Interpretation of the Quantum Theory in Terms of Hidden Variables I," Phys. Rev., vol. 85, pp. 166-179, 1952.
- [16] D. Bohm, "A Suggested Interpretation of the Quantum Theory in Terms of Hidden Variables II," Phys. Rev., vol. 85, pp. 180-193, 1952.
- [17] J. S. Bell, "On the Einstein-Podolsky-Rosen paradox," Physics, vol. 1, pp. 195-200, 1964.
- [18] J. S. Bell, "On the Problem of Hidden Variables in Quantum Mechanics," Rev. Mod. Phys., vol. 38, pp. 447-452, 1966.
- [19] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information, Cambridge, 2000.
- [20] E. Knill, "Conventions for Quantum Pseudocode," citeseerx, 1996.
- [21] E. Knill, "Conventions for quantum pseudocode. 1996.
- [22] Marco Lanzagorta and Jeffrey Uhlmann, Quantum Computer Science, 1st ed. Cambridge University Press, 2009.
- [23] Bernhard Omer, "Structured Quantum Programming," Master's thesis, Institute for Theoretical Physics Vienna University of Technology, 2003.
- [24] Bernhard Omer, "Structured quantum programming," Information Systems, 2005.
- [25] "A Functional Quantum Programming Language," 20th Annual IEEE Symposium on Logic in Computer Science, Chicago, Illinois, 2005.
- [26] Hynek MLNAÍK, "Semantics of Quantum Programming Language LANQ," International Journal of Quantum Information, vol. 6, pp. 733-738, 2008.
- [27] Han-Cheng Shih, Kuo-Chang Lee, and Tzonelih Hwang, "New Efficient Three-Party Quantum Key Distribution Protocols," IEEE, vol. 15, pp. 1602-1606, 2009.
- [28] A. S. Tanenbaum and A. S. Woodhull, Operating Systems Design and Implementation, 3rd ed. Pearson Prentice Hall, 2006.
- [29] Song, Fang-Min | Qian, Shi-Jun Dai, Jing-An Zhang, Yun-Jie Xu and Jia-Fu, "Processing system of quantum programming language NDQJava," Ruan Jian Xue Bao, vol. 19, pp. 9-16, 2008.
- [30] Joshua Bloch, Effective Java, 2nd ed. Addison-Wesley, 2008.
- [31] Simon J. Gay, "Quantum programming languages: survey and bibliography," Mathematical Structures in Computer Science, vol. 16, pp. 581-600, 2006.

اندازه‌گیری، بیشتر از مدت زمان ناهمدوسی باشد نتایج به دست آمده از پردازش اعتبار ندارد.

تاکنون بررسی‌های مفصلی بر روی ساختارهای داده‌ای پیچیده کوانتومی و ساختارهای پیچیده کنترل کوانتومی در زبان‌های برنامه‌سازی انجام نشده است و عموماً از ساختارهای ساده داده‌ای و کنترلی استفاده کرده‌اند. تحقیق در این زمینه می‌تواند ابزارهای مناسبی را برای ایجاد برنامه‌های کوانتومی پیچیده‌تر و کارآتر، فراهم کند.

زمانی که ساختار زبان‌های برنامه‌سازی پیچیده می‌شود، مانند زبان‌های برنامه‌سازی ساخت‌یافته و شیء‌گرا، بررسی معنایی آن‌ها هم به نوبه خود کاری پیچیده خواهد بود. زبان‌هایی که به درستی از نظر معنایی مورد بررسی قرار گرفته‌اند ساختارهای جبری داشته‌اند، از این رو نمی‌توانند برای برنامه‌سازی مناسب باشند. ایجاد ابزارها و روش‌های مناسب جهت بررسی معنایی زبان‌های برنامه‌سازی امری ضروری است که باید دیر یا زود به آن پرداخته شود.

هنوز روش موثری برای ایجاد یک پردازنده موثر کوانتومی شناخته نشده است و حتی توانایی ایجاد یک پردازنده کوانتومی در پرده ابهام است. اما با این وجود فناوری‌های متفاوتی که در پیاده‌سازی پردازنده‌های کوانتومی مورد استفاده قرار گرفته‌اند، هر کدام دسته‌ای از دستورها را معرفی کرده‌اند که به راحتی می‌تواند با کارایی مناسبی بر روی پردازنده کوانتومی ایجاد شده پیاده‌سازی شود. از سویی هر کدام دارای قابلیت‌های متفاوت و زمان‌های ناهمدوسی متفاوتی نسبت به بقیه هستند، گرچه در حالت کلی می‌توان نشان داد که عموماً از یک سری دستورهای پایه که شباهت زیادی با هم دارند استفاده می‌کنند. ایجاد مترجم‌هایی که بتواند بر اساس ساختار فیزیکی خاص، برنامه‌های سطح بالا را ترجمه کند بسیار مفید است. یکی از نیازهای دیگر که به دنبال ایجاد مترجم بر اساس فناوری خاص به وجود می‌آید، ایجاد مترجم‌هایی است که بتواند یک برنامه را که به دستورات یک سیستم خاص ترجمه شده، به برنامه‌ای برای یک سیستم دیگر ترجمه کند. البته بررسی درستی ترجمه‌های انجام شده و بهینه بودن آن‌ها نیز پرسش‌هایی اساسی است که در آینده این فرآیندها مطرح خواهد شد.

منابع

- [1] C.H. Bennett and G. Brassard, "Quantum Cryptography: Public Key Distribution and Coin Tossing," Computers, Systems, and Signal Processing, pp. 175-179, 1984.
- [2] M. A. Nielsen and Isaac L. Chuang, "Programmable Quantum Gate Arrays," Phys. Rev. Lett., vol. 79, pp. 321-324, 1997.
- [3] P. W. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," 35th Annual Symposium on Foundations of Computer Science, pp. 124-134, Santa Fe, NM, USA, 1994.
- [4] David Deutsch and Richard Jozsa, "Rapid Solution of Problems by Quantum Computation," Proc. R. Soc. Lond. A, vol. 439, no. 190, pp. 553-558, 1992.
- [5] Lov K. Grover, "Quantum Mechanics Helps in Searching for a Needle in a Haystack," Phys. Rev. Lett., vol. 79, pp. 325-328, 1997.

شانزدهمین همایش بین‌المللی هوش مصنوعی و پردازش سیگنال



تشکر و قدردانی

با معرفی آقای دکتر جعفر حبیبی، رئیس هیأت مدیره انجمن کامپیوتر ایران، و موافقت و همکاری آقای دکتر فرشاد خونجوش، دبیر همایش، سه نفر از اعضای شاخه دانشجویی انجمن انفورماتیک ایران به نام‌های خانم‌ها مریم دانیالی، سپیده علوی و مهدیه محسنی به طور رایگان در این همایش شرکت کردند که بدین وسیله از آقای دکتر حبیبی و آقای دکتر خونجوش صمیمانه قدردانی می‌گردد.

در تاریخ سیزدهم و چهاردهم اردیبهشت ماه ۱۳۹۱، شانزدهمین همایش بین‌المللی هوش مصنوعی و پردازش سیگنال از سوی انجمن کامپیوتر ایران و با همکاری دانشگاه شیراز در محل این دانشگاه برگزار شد.

در روز افتتاحیه، ابتدا آقای دکتر مرزوقی، معاون آموزشی دانشگاه شیراز، ضمن خوشامدگویی به حضار، همایش را افتتاح کرد و ابراز امیدواری نمود که نتایج و دستاوردهای این همایش وارد حوزه آموزشی شود.

سپس آقای دکتر موقر، عضو هیأت علمی دانشکده مهندسی کامپیوتر دانشگاه صنعتی شریف و نماینده انجمن کامپیوتر ایران به ایراد سخنرانی به دو زبان فارسی و انگلیسی پرداخت و در رابطه با فعالیت‌های علمی انجمن کامپیوتر ایران مطالبی را به اطلاع حضار رساند.

آخرین سخنران مراسم افتتاحیه، آقای دکتر فرشاد خونجوش، دبیر همایش و عضو هیأت علمی دانشکده مهندسی کامپیوتر دانشگاه شیراز بود. وی هدف از برگزاری این همایش را ایجاد محلی برای تبادل اطلاعات بین متخصصان فن عنوان کرد. آقای دکتر خونجوش همچنین به معرفی فعالیت‌های علمی و پژوهشی دانشکده مهندسی کامپیوتر دانشگاه شیراز در زمینه‌های هوش مصنوعی، مهندسی نرم‌افزار و پردازش موازی پرداخت و به همکاری این دانشکده با بخش صنعت اشاره کرد.

پس از خاتمه سخنرانی‌های مراسم افتتاحیه، برنامه همایش با ارائه شفاهی مقالات پذیرفته شده در دو بخش دنبال شد. ساعتی نیز به ارائه پوسترها و پرسش و پاسخ از ارائه‌کنندگان آن‌ها اختصاص یافته بود.

از مهمانان خارجی حاضر در همایش می‌توان به دکتر سعید صانعی، استاد دانشگاه سوریلندن اشاره کرد. پروفسور کریستین یوتن، استاد دانشگاه ژوزف فوریه فرانسه نیز به صورت ویدئو کنفرانس سخنرانی خود را ارائه کرد.

در این همایش از نخستین آنتی‌ویروس ملی کشور به نام «آپا» با حضور خانم دکتر نسرین سلطان‌خواه، معاون علمی و فناوری ریاست جمهوری رونمایی شد.

در پایان همایش از مقالات برتر قدردانی شد.

تحقق شهر الکترونیکی: یک نگاه اجمالی در پدیده‌شناسی از منظر رویکرد تحول

اردوان مجیدی

موسسه مطالعات راهبردی فناوری اطلاعات

پست الکترونیکی: majidi.office@gmail.ir

رضا درخشان

موسسه مطالعات راهبردی فناوری اطلاعات

پست الکترونیکی: Derakhshan.r@gmail.com

چکیده

می‌گیرد. در این مقاله در شناخت پدیده شهر الکترونیکی، پدیده‌های جامعه، اطلاعات، فناوری اطلاعات، رایا سپهر، جامعه اطلاعاتی و شهر مورد بررسی و تحلیل قرار می‌گیرند. سپس بر اساس این تحلیل، خصوصیات شهر الکترونیکی تبیین شده؛ و در نهایت بر اساس تحلیل ابعاد پدیده‌های مذکور و خصوصیات شهر الکترونیکی، خصوصیات رویکرد تحول در تحقق شهر الکترونیکی مورد بحث قرار می‌گیرد. از منظر این خصوصیات، پدیده شهر الکترونیکی، نوع جدیدی از شهر، و در بردارنده نوع جدیدی از جامعه است، و تنها شامل به‌کارگیری برخی ابزارها نیست. طرح‌های توسعه شهر الکترونیکی، باید در منظر این نگاه، پاسخگوی نیازهای تحولی شهر باشند. با توجه به ابعاد وسیع پدیده‌های مذکور، این پدیده‌شناسی تنها در محدوده نیازمندی‌های کلیدی رویکرد تحول صورت می‌گیرد.

کلمات کلیدی

شهر الکترونیکی، جامعه، اطلاعات، فاوا، سایبرنتیک (رایانیک)، رایا سپهر، جامعه اطلاعاتی، جامعه تمدنی، شهر، رویکرد تحول، پدیده‌شناسی

طراحی، برنامه‌ریزی و توسعه شهرها در شرایط شکل‌گیری جامعه اطلاعاتی، علاوه بر مد نظر قراردادن ابعاد کالبدی، نیاز به پوشش ابعاد دیگری نیز هست؛ که نگاه به پدیده شهر در بعد رایا سپهری^۱ آن، بخصوص در تحقق شهر الکترونیکی، از آن جمله است. آنچه که در تحقق شهر الکترونیکی دارای اهمیت است، مشخص ساختن رویکردی مناسب است که تحول شهر به سمت شهر الکترونیکی را با سازوکار و صورت مناسب، امکان‌پذیر سازد. اتخاذ رویکرد تحول مناسب، بیش از هر چیز نیازمند شناخت پدیده شهر الکترونیکی است. ابعاد مختلف این شناخت، به ما کمک می‌کند تا خصوصیات و ملزومات تحقق شهر الکترونیکی را تعیین نموده، و بر اساس آن معماری شهر الکترونیکی، و نیز برنامه توسعه شهر الکترونیکی را ترسیم کنیم. منظور از معماری در اینجا، نه معماری فضای کالبدی، بلکه معماری رایا سپهری، سازمانی و اجتماعی در پهنه جامعه اطلاعاتی است. از این رو رویکرد معماری سازمانی و معماری نظام‌های کلان، در این شناخت مورد استفاده قرار

۱- مقدمه

حمد، خداوند متعال را که برترین موجودات، و خلیفه خود در زمین را چنان آفرید که با جمع شدن در کنار یکدیگر، پدیده جامعه را به گونه‌ای شکل می‌دهند که روح و رفتار مستقلی را به عنوان یک موجود متفاوت دارا است و در پس خلق تک موجود متعالی که روح او در آن دمیده شده، خلقی دیگر، در ترکیب و به هم پیوستن این تک موجودات نهفته است. و تکامل و تعالی این خلق دیگر، در یافتن انسجام و یکپارچگی در قالب یک نظم تمدنی که به صورت جغرافیائی و محیطی سازمان یافته است، و جریان یافتن سازوکارهای زندگی و معیشت در بستر آن، در قالب پدیده شهر است که خود خلقی دیگر است.

و تجلی زیبایی این خلق احسن زمانی بیشتر می‌شود که این موجود، با ساماندهی نوعی از خودکارسازی، خودسازماندهی، و مدیریت بازخوردهای آن در بهبود عملکرد و سازماندهی این پدیده شهر به شکل رایا سپهری، با بهره‌گیری از برترین و زیباترین آفریده‌های خود در این دوران، یعنی فناوری اطلاعات، بستر نوع دیگری از این خلق دیگر را پی‌ریزی کند؛ شهر الکترونیکی!

در تبیین ابعاد و چگونگی تحقق شهر الکترونیکی، طراحی معماری شهر الکترونیکی و نیز برنامه توسعه شهر الکترونیکی، نقش اساسی را ایفا می‌کنند. در اینجا لازم است منظور ما از به کاربردن واژه معماری، به صورت شفاف مورد توجه قرار گیرد. هر چند اصطلاح "معماری" عمدتاً در حوزه طراحی سازه‌های فیزیکی مانند ساختمان، شهر و نظایر آن، و بخصوص در تبیین ابعاد کالبدی آن به کار می‌رود، ولی این اصطلاح در سالیان اخیر با الهام گرفتن از حوزه معماری ساختمان، در حوزه‌های دیگری رواج پیدا کرده و به کار گرفته می‌شود. از این جمله می‌توان به معماری رایانه، معماری نرم‌افزار، معماری سازمان، و نیز معماری نظام‌های کلان (نظیر نظام اقتصادی، نظام سلامت، نظام آموزش)، اشاره نمود.

معماری یک نظام کلان، تبیین یک نگاه کل‌گرا و عقلانی بر یک نظام کلان پیچیده است که اجازه تمرکز بر مولفه‌های کلیدی و هم‌بندی و تعامل آن‌ها را می‌دهد و امکان پرهیز از ورود به جزئیات را فراهم می‌کند [مجیدی ۸۵-۲]. یک شهر، از یک منظر یک سازه فیزیکی است و در قالب مسائل کالبدی و نیز ارتباط مسائل اجتماعی، روانی، فرهنگی، هنری و سیاسی با این مسائل کالبدی، در چارچوب علم، فن و هنر معماری ساختمان و شهرسازی مورد تحلیل و طراحی قرار می‌گیرد. از طرف دیگر یک شهر یک نظام کلان محسوب می‌شود و می‌توان از منظر مهندسی و معماری نظام‌های کلان، به آن و نظام‌های آن نگاه کرد. بخصوص در مورد شهر الکترونیکی، با توجه به این که بسیاری از مسائل آن از منظر رایا سپهری مورد توجه و تحلیل قرار می‌گیرند، زمینه تحلیلی بیشتری را در حوزه مهندسی و معماری نظام‌های کلان پیدا می‌کند. لذا ما نیز در این مقاله از دیدگاه

معماری سازمان و معماری نظام‌های کلان، به بحث معماری و تحول شهر الکترونیکی نگاه می‌کنیم.

یکی از مسائل گلوگاهی که در حوزه مهندسی و معماری نظام‌های کلان مورد توجه قرار دارد، رویکرد تحول نظام کلان مربوطه است. از همین منظر، از مهمترین عواملی که در تحقق شهر الکترونیکی دارای اهمیت است، مشخص ساختن رویکردی مناسب است که تحول شهر به سمت شهر الکترونیکی را با سازوکار و صورت مناسب، امکان‌پذیر سازد. اتخاذ رویکرد تحول مناسب، بیش از هر چیز نیازمند شناخت پدیده شهر الکترونیکی است. ابعاد مختلف این شناخت به ما کمک می‌کند تا خصوصیات و ملزومات تحقق شهر الکترونیکی را تعیین نموده، و بر اساس آن معماری شهر الکترونیکی و نیز برنامه توسعه شهر الکترونیکی ترسیم شود. تا ما به درستی نشناسیم که پدیده شهر الکترونیکی، چگونه پدیده‌ای است و خصوصیات آن از زوایای مختلف چگونه است، نمی‌توانیم معیاری برای سنجش صحت معماری داشته باشیم. و تا معیاری برای سنجش معماری نداشته باشیم، معیاری برای سنجش صحت برنامه توسعه نیز نخواهیم داشت. به عبارت دیگر، بر اساس مشخصه‌هایی که پدیده‌شناسی از پدیده شهر الکترونیکی پیش روی ما قرار می‌دهد، طراحی معماری به گونه‌ای صورت می‌گیرد که این مشخصه‌ها تحت پوشش قرار گرفته، و ملزومات و نیازمندی‌های شهر الکترونیکی توسط معماری تامین و پاسخ داده شود. آنگاه برنامه توسعه نیز بر اساس ترسیمی که معماری از سازوکارها و ساختار و مولفه‌ها و هم‌بندی آن ارائه می‌کند، مسیر راهبردی و عملیاتی تحول شهر به سمت شهر الکترونیکی ترسیم شده و تحقق آن را مشخص خواهد نمود.

برای دست یافتن به خصوصیات و ابعاد یک پدیده، شناخت پدیده‌های کلیدی شکل دهنده آن پدیده، خصوصیات این پدیده‌ها و هم‌بندی آن‌ها در اثرگذاری متقابل خصوصیات آن‌ها بر یکدیگر، نقش اساسی را ایفا می‌کند. از آنجا که شهر الکترونیکی اولاً شکلی از یک جامعه است، لازم است تا مفهوم جامعه، آن هم با شکل تحول یافته و نوین آن یعنی جامعه اطلاعاتی، مورد اشاره و تعریف و تبیین مفهوم قرار گیرد. ضمن آن که برای تبیین پدیده جامعه اطلاعاتی، لازم است تا پدیده اطلاعات، فناوری اطلاعات و رایا سپهر، که یک محور اساسی جامعه اطلاعاتی را تشکیل می‌دهند، مورد بحث قرار گیرند. در این میان با توجه به نقش اساسی فرهنگ در شکل‌گیری جامعه، و نقش آن در تحول نظام، پدیده فرهنگ و منظر نگاه ما به این پدیده، مورد اشاره اجمالی قرار می‌گیرد. همچنین لازم است تا پدیده شهر، به عنوان شکل خاصی از ظهور جغرافیائی جامعه متمدن، به اجمال معرفی شود. پس از آن می‌توان به پدیده شهر اطلاعاتی پرداخت و با توجه به پدیده‌شناسی اجمالی مولفه‌ها و پدیده‌های سازنده آن، خصوصیات اساسی این پدیده را مورد بحث مختصر قرار داد. در نهایت بر اساس این خصوصیات تبیین شده، می‌توان به خصوصیات رویکرد تحول در تحقق شهر الکترونیکی اشاره کرد.

با توجه به این که پدیده‌های مورد توجه در این پدیده‌شناسی، دامنه وسیع و ابعاد متعددی دارند، برای محدود ساختن دامنه و امکان جمع نمودن ابعاد پدیده‌شناسی در این نوشتار، لازم است تا تنها ابعادی مورد توجه بیشتر قرار گیرد که اثر مشخصی در تعیین ابعاد رویکرد تحول خواهند داشت. به همین جهت، این مقاله تنها یک پدیده‌شناسی از منظر رویکرد تحول خواهد بود.

این پدیده‌شناسی بر اساس مطالعاتی که در پروژه‌های طراحی معماری و تدوین سند ملی دولت الکترونیکی [سند دولت الکترونیکی ۸۴]، معماری فاوای نهاد ریاست جمهوری [معماری فاوای نهاد ۸۴]، معماری سازمان فرهنگی هنری شهرداری تهران [معماری سازمان فرهنگی ۸۶]، و چارچوب و متدولوژی معماری سازمانی (چم) [مجیدی ۸۵-۲] انجام شده، شکل گرفته و تکامل یافته است.

۲- جامعه

با توجه به این که پدیده شهر الکترونیکی، در هر صورت نوعی از جامعه است، لازم است تا زاویه نگاه ما به پدیده جامعه مشخص شود. تعاریف مختلفی از پدیده جامعه، از زوایای مختلف وجود دارد. برخی از این تعاریف عبارتند از:

جامعه، از سوی جامعه‌شناسان و انسان‌شناسان، به عنوان مجموعه روابط اجتماعی بین افراد تعریف شده است. جامعه، گروهی از افراد است که به طور وسیع با گروه‌های دیگر از نظر منافع مشترک، روابط مشخص، نهادهای اشتراکی، و فرهنگ مشترک، متمایز می‌شوند.

[Srinivasan 2004]

جامعه، راهی است که افراد خود را در آن آرایش و سازماندهی می‌کنند. از این رو، می‌توان به تنوع گروه‌ها، شبکه‌ها، نهادها، سازمان‌ها و سیستم‌ها که افراد از آن‌ها برای ارتباط خود با یکدیگر برای شکل‌دهی یک جامعه استفاده می‌کنند توجه کرد.

[Howitt 2009]

جامعه، از یک عده افرادی تشکیل شده است که برای ارضای نیازهایشان، با هم روابط اجتماعی دارند. این تعریف یک انتظار ایده‌آل از کارکرد ایده‌آل یک جامعه است و بیشتر جامعه‌شناسان کارکردگرا به آن اعتقاد دارند. [ارفع‌پور ۸۲]

با نگاهی به تعاریف موجود، تبیین مفهوم جامعه می‌تواند مبتنی بر شمارش ۱۲ رکن اساسی آن صورت گیرد. مبتنی بر این ۱۲ رکن، یک جامعه عبارت است از:

- مجموعه‌ای از انسان‌ها، و نیز افزارها، موجودات زنده و منابع تحت انقیاد آن‌ها،
- در یک نظام مشخص پایدار (و نه لزوماً مدون)،
- سازماندهی شده در سازمان‌های رسمی و غیررسمی،
- در محدوده جغرافیائی و محیط مشخص،

- در رابطه و تعامل با یکدیگر و محیط،
- که دارای فرهنگ مشخص شکل‌یافته طبق مجموعه‌ای از رفتارها،
- و نیز گستره‌ای از برخی از وجوه مشترک،
- که افراد در آن، با درک وجود و رفتار خود و دیگران و محیط، رفتار بامعنائی را تنظیم، شکل داده و بروز می‌دهند،
- و کل آن جامعه نیز به مثابه یک موجود و پدیده زنده و دارای روح واحد،
- رفتار یکپارچه بامعنائی را به صورت خودآگاه یا ناخودآگاه، از خود بروز می‌دهد،
- و روح و رفتارهای افراد و جامعه، قوه محرکه کافی را برای تحول درونی و خود سازمانده ساختار و رفتار درونی و رفتار بیرونی، ایجاد می‌کند.

در یک تعریف چکیده، جامعه، مجموعه‌ای از انسان‌ها در یک نظام مشخص پایدار، در رابطه و تعامل بامعنا با یکدیگر و محیط است که دارای فرهنگ مشخص و گستره‌ای از وجوه مشترک، روح واحد و خود سازمانده است.

جامعه مانند یک موجود زنده دارای روح است، از این رو، رفتار یکپارچه و بامعنائی را از خود نشان می‌دهد. همچنین، یک بستر و فضای تصمیم‌گیری مشخصی در جامعه وجود دارد که تصمیم‌گیری کل جامعه در آن نقش دارد. در این بستر و فضا، همه افراد موجود در جامعه، به شیوه‌های مختلفی در رفتار آن دخالت دارند. هر شخصی با ایفای نقش خود در جامعه، می‌تواند منجر به اتخاذ یک تصمیم عمومی در جامعه شود. این بستر و فضای تصمیم‌گیری، لزوماً مشابه با بستر و فضای تصمیم‌گیری که در علوم مدیریت، سیاست و مهندسی تصمیم وجود دارد، مستلزم انتخاب خودآگاه و مبتنی بر اعلام افراد نیست. چنین نیست که برای گرفتن یک تصمیم، همه افراد جامعه کنار هم جمع شده و تصمیمی را اتخاذ کنند. به عنوان مثال شخصی را برای انجام کاری انتخاب کنند. در بیشتر تصمیم‌گیری‌هایی که در بطن جامعه اتفاق می‌افتد، اعلام تصمیم وجود ندارد و تنها رفتار، بیانگر آن تصمیم‌گیری است. جامعه تصمیم می‌گیرد که در آن شرایط چه رفتاری را داشته باشد. بنابراین، تصمیم‌هایی که جامعه می‌گیرد، به شکل اعمال هنجارها و شکل‌گیری رفتارها بروز می‌کند. از این رو، نوع تصمیم‌گیری جامعه، نوع سازماندهی آن، و مراحل که برای اتخاذ آن در جامعه انجام می‌شود و خروجی آن، متفاوت با باورهای رایجی است که در مورد مساله تصمیم‌گیری وجود دارد. بستر و فضای تصمیم‌گیری هر فرد، مجموعه‌ای است از باورها و عقاید، رفتارها، و محیطی که فرد در آن واقع شده است، که موجب می‌شود در شرایطی خاص، جریان تصمیم‌گیری به گونه‌ای هدایت شود که او یک تصمیم مشخصی را اتخاذ نماید. افراد مختلف، دارای فضاهای تصمیم‌گیری متفاوت و مخصوص به خود هستند و باورها و اعتقادات، رفتارها، و مسائل محیطی خاص آن‌ها سبب می‌شود که اگر دو نفر

هر چند که ابعاد مختلف بحث در این میدان بسیار وسیع، و دستیابی به پاسخ واحد و فراگیر بسیار دشوار است، اما در این پدیده‌شناسی لزوم ارائه یک پاسخ واحد، و تعریف مشخص از این پدیده، برای دستیابی به یک تفاهم فکری بنیادین، در بافت اجرائی و عملیاتی میدان پژوهش و برنامه‌ریزی مذکور ضروری است. برخی از تعاریف ارائه‌شده از فرهنگ توسط صاحب‌نظران عبارتند از:

- فرهنگ عبارت است از ارزش‌هایی که اعضای یک گروه معین دارند، هنجارهایی که از آن پیروی می‌کنند و کالاهای مادی که تولید می‌کنند. فرهنگ به شیوه زندگی اعضای یک جامعه معین - عادات و رسوم آن‌ها، همراه با کالاهای مادی که تولید می‌کنند - مربوط می‌شود. [گیدنز ۸۳]
- فرهنگ، مجموعه منسجمی است از ابزار و وسایل و اجناس مصرفی، منشورها و قوانین اساسی برای گروه‌های اجتماعی، اندیشه‌ها و پیشه‌ها، باورها و آداب و رسوم. [مالینوفسکی ۸۳]
- از نظر مویل‌مان که یک انسان‌شناس فرهنگی است، فرهنگ، کلیه اشکال زندگی یک قوم، از جمله پایه‌های تفکری آن را تشکیل می‌دهد و این همچنین شامل تمام زمینه‌های فنی موجودیت آن قوم از جمله البسه، ساختمان، ابزار، و ... نیز می‌گردد. [رفیع‌پور ۸۲]

با تمایل شدید به این زاویه نگاه که فرهنگ جامعه را باید در فضای شکل‌دهنده رفتار جامعه جستجو کرد، و نه تنها آداب و رسوم و امثال آن، فرهنگ را این چنین تعریف می‌کنیم:

"فرهنگ بستری دربردارنده قابلیت "ادراک، شناخت و قدرت طبقه‌بندی و تفکیک پدیده‌های مختلف و طیف‌های مختلف آن‌ها"، به عنوان محمل شکل‌گیری "ارزش‌ها و تحقق خواست‌ها و اراده"، با تولید و تاثیرپذیری متقابل از فرارفتارهایی نظیر "باورها، ایدئولوژی، عادات، مهارت‌ها، هنجارها"، و فرسازه‌هایی نظیر "مفاهیم، نمادها، ضرب‌المثل‌ها، اسطوره‌ها، نهادها و ابزارها"، در بروز رفتارهای بامعنا باثبات فرد، جامعه، حکومت و دولت، به صورت خودآگاه یا ناخودآگاه، که متضمن راه و رسم زندگی یا فعالیت کردن در یک محیط مشخص، و رشد، پرورش و تکامل آن است."

و در تعریفی چکیده:

"فرهنگ بستری متضمن راه و رسم زندگی کردن، و پرورش آن، برای بروز رفتارهای بامعنا باثبات بشر است که دربردارنده قابلیت شناخت، به عنوان محمل شکل‌گیری ارزش‌ها، با تولید و تاثیرپذیری متقابل از فرارفتارها و فرسازه‌ها است."

فرارفتار عاملی است که سبب می‌شود تا رفتار مشخصی در مقابل یک طیف یا سطح خاصی از پدیده‌ها شکل گیرد. فرارفتار، مشخص می‌کند که چه رفتاری، در چه شرایط و در چه خصوصیتی، باید بروز کند و هنگامی که خصوصیات و شرایط طیفی از پدیده تحقق پیدا کرد، انسان را به طور خودآگاه یا ناخودآگاه، به رفتار بامعنائی در

در شرایط مشابهی قرار گیرند، تصمیمات مختلفی را اتخاذ نمایند. بستر و فضای تصمیم‌گیری شخص، با بستر و فضای تصمیم‌گیری کل جامعه، دارای ارتباط تنگاتنگی است. بستر و فضای تصمیم‌گیری هر شخص، در درون بستر و فضای تصمیم‌گیری جامعه قرار دارد و تصمیماتی که شخص می‌گیرد، در بستر و فضای تصمیم‌گیری جامعه اثر می‌گذارد و برعکس. از این رو، اثرگذاری متقابلی در این میان وجود دارد. قوای محرکه اصلی جامعه، درونی‌شدن تصمیم‌گیری در وجود عموم افراد جامعه و کل جامعه است. رفتارهایی که جامعه از خود نشان می‌دهد، در درون تک تک افراد جامعه (و یا عموم افراد) درونی می‌شود و تصمیم‌هایی که در جامعه اتخاذ می‌شود، در بستر و فضای تصمیم‌گیری تک تک افراد (و یا عموم افراد) شکل می‌گیرد. یک تصمیم جامعه که منجر به یک رفتار مشخص در کل جامعه می‌شود، توسط آحاد جامعه و در تصمیم‌گیری و رفتار تک تک (نه همه) آن‌ها، به صورت خودآگاه و یا ناخودآگاه شکل می‌گیرد. [مجیدی ۸۵-۱]

۳- فرهنگ

در طراحی معماری و برنامه‌ریزی تحول شهر الکترونیکی، پدیده فرهنگ به عنوان یک بستر سازنده رفتارهای جامعه، نقش جدی را ایفا می‌کند. تحقق شهر الکترونیکی مستلزم نهادینه شدن آن در فرهنگ جامعه است. تجربیات متعدد نشان داده است که کندی یا سرعت پیشرفت و تکامل رایا سپهرها، عملاً تنها در گرو کندی یا سرعت پیشرفت فناوری یا منابع اقتصادی نیست، بلکه بسیار بیش از آن‌ها در گرو تحول فرهنگ است. در اغلب موارد، فناوری بسترهای الکترونیکی که امروزه در یک جامعه متداول است، چندین سال قبل وجود داشته است؛ منابع اقتصادی برای به کارگیری آن نیز وجود داشته است؛ حتی دانش فنی و نیروی متخصص نیز وجود داشته است؛ اما به دلیل عدم پذیرش فرهنگی و شکل‌گیری فرهنگ استفاده و ابعاد فرهنگی جنبی استفاده از آن، تحقق آن بسترها سال‌ها به تاخیر افتاده است. نمونه‌های این مشکل را امروزه در بسیاری از موارد نظیر بهداشت الکترونیکی، دولت الکترونیکی و آموزش الکترونیکی مشاهده می‌کنیم. شهر الکترونیکی نیز از این موضوع مستثنی نیست. تحقق شهر الکترونیکی ابتدائاً نیازمند شکل‌گیری فرهنگ شهر الکترونیکی خواهد بود.

آیا فرهنگ تنها به حوزه مسائلی نظیر آداب و رسوم، هنر بومی، گویش‌ها و زبان، آثار باستانی و نظایر آن محدود می‌شود؟ آیا مولفه‌های سازنده فرهنگ را تنها در این حوزه از مسائل باید جستجو نمود؟ یا آنچه که سبب شکل‌گیری رفتارهای باثبات و با معنای فرد و جامعه در برخورد با پدیده‌های محیط اطراف می‌گردد، فرهنگ محسوب می‌شود؟

مقابل آن پدیده هدایت می‌کند. باورها، هنجارها، عادت‌ها، مهارت‌ها، از انواع فرارفتار محسوب می‌شوند.

یک فرهنگ، طی دوران عمر خود، برخی از فرارفتارها را در برخی از مصنوعات، متجلی می‌کند به نحوی که این مصنوع، به گونه‌ای آن فرارفتارها را نمایان کرده و استفاده کننده از این مصنوع را به اتخاذ آن فرارفتار وادار می‌کند. چنین مصنوعی را "فرسازه" می‌نامیم. مثلاً وقتی یک فرهنگ، توسط نوع خاصی از پوشش، غذا، یا ابزار، رفتاری را نهادینه می‌کند. یک فرسازه، سازه‌ای است که رفتارهای خاصی را به صورت یک بسته، در بر دارد. و جامعه با ایجاد آن، مولفه‌هایی از فرهنگ را ظهور بخشیده، و امکان انتقال یا ذخیره‌سازی آن را فراهم می‌آورد.

بر اساس این تعریف، شکل‌گیری یک بستر یا فضا نظیر شهر الکترونیکی، مستلزم شکل‌گیری فرارفتارها و فرسازه‌های مناسب آن در جامعه، به صورتی است که رفتارهای باثبات و مورد انتظار در آن بستر یا فضا، به صورت مناسب توسط آحاد جامعه بروز پیدا کند.

۴- اطلاعات، فناوری اطلاعات، رایا سپهر

بر اثر به کارگیری وسیع سیستم‌های رایانه‌ای، تغییرات در ۵ لایه: ۱- کاربردهای مستقیم سیستم‌های رایانه‌ای، ۲- روش‌های مدیریت، ۳- مهندسی سیستم، ۴- اصول، راهبردها، ماموریت و سیاست‌ها، و ۵- مفاهیم و نظریات علمی منتشر می‌شود. آنچه در اینجا بخصوص در لایه‌های چهارم و پنجم مطرح است، دیگر سیستم‌های رایانه‌ای نیست، بلکه نوعی جدید از به کارگیری و نگرش به ساختارهای اطلاعاتی، و مفهومی جدید در حال شکل‌گیری است. نوعی جدید از کنترل بشر بر ساختارهای اطراف خود بر اثر این به کارگیری ایجاد می‌شود. و این کنترل را شاید بتوان در چارچوب واژه اطلاعات از یک سو در منظر ماهیت کنترل، فناوری اطلاعات از سوی دیگر در منظر بستر ادراک سازوکار، و سایبرنتیک (رایانیک) را در سوی دیگر از منظر تحلیل رفتار، مورد توجه قرار داد.

اطلاعات، تبیین مجموعه‌ای از حقایق و واقعیت‌های پدیده‌ها (نه لزوماً همه حقایق و واقعیت‌های پدیده‌ها)، در قالب عناصر مفهومی است که به شکل داده‌ها، اطلاعات، دانش یا علم، نمایان می‌شود و منجر به شکل‌گیری آگاهی و بصیرت انسان در درک معنای آن حقایق و واقعیت‌های پدیده‌ها می‌شود.

"فناوری اطلاعات و ارتباطات، بستر نظری، سیستمی و ابزاری برای درک مفاهیم و بنیان‌های نظری، سازماندهی، کنترل و هدایت راهبردی در نظام‌های خرد و کلان بهره‌مند از به کارگیری اساسی سیستم‌های رایانه‌ای اطلاعاتی و ارتباطی است" [مجیدی، ۸۳]

سایبرنتیک (رایانیک) علم خودکار سازی در بهبود عملکرد و سودمندی اثر فعالیت یک نظام (طبیعی یا مصنوعی) است که به صورت جدی از مطالعه بازخوردهای رفتارها و اثرات آن‌ها، بهره می‌برد. در ادبیات

رایج علمی مجموعه بسترها، عناصر و مولفه‌های سازنده یک نظام که به صورت موثر از سازوکارهای این علم به صورت مصنوعی بهره می‌برند، رایا سپهر نامیده می‌شود.

فناوری اطلاعات و ارتباطات ابزار نیست. مستقیماً یک نظریه علمی هم نیست. روش هم نیست. اما همه این‌ها را به نوعی حمایت می‌کند. فاوا فقط محملی را برای توجیه و توضیح اتفاقاتی که در حال رخ دادن است و مسائلی که در حال شکل‌گیری در رایا سپهر است، بخصوص در لایه چهارم و پنجم مورد بحث، فراهم می‌کند. حتی فاوا یک رشته علمی مشخص هم نیست. همه علوم به نوعی به تدریج به محدوده این بستر وارد می‌شوند و مسائل خود را در محدوده آن طرح و تبیین می‌کنند. برای همین مسئله است که ما شاهد شکل‌گیری واژه‌هایی نظیر e-learning، e-government، e-health، e-commerce و نظایر آن، و از جمله e-city هستیم.

هر چند واژه الکترونیک، به ابزارها و علم و فناوری‌های مرتبط در حوزه این افزارها اطلاق می‌شود، اما به کارگیری این واژه در واژه‌هایی نظیر تجارت الکترونیک، دولت الکترونیک، و نظایر آن‌ها را می‌توان با تسامح (در وجه معنا و توجه افزاری آن)، به معنا و مفهوم مبتنی‌شدن وسیع‌الطیف آن حوزه به اطلاعات و ارتباطات، نظام‌های اطلاعاتی و فناوری اطلاعات و ارتباطات، و شکل‌گیری آن در رایا سپهر دانست. از این رو، تجارت الکترونیک مترادف تجارت مبتنی بر اطلاعات و ارتباطات و نظام‌های اطلاعاتی و فناوری اطلاعات در رایا سپهر است، و شهر الکترونیک نیز مترادف شهر مبتنی بر اطلاعات و ارتباطات و نظام‌های اطلاعاتی و فناوری اطلاعات در رایا سپهر خواهد بود.

در واقع این واژه‌ها در حال اعلام این مسئله هستند که نوع جدیدی از آموزش، دولت، بهداشت، تجارت، ... و البته شهر در حال شکل‌گیری است. این‌ها نه تنها شیوه کاری جدید هستند، بلکه علمی جدید نیز هستند که نظریات و مفاهیم علمی جدیدی در آن‌ها مطرح می‌شود. و تفکیک بین علم آموزش (learning) و علم آموزش الکترونیک (e-learning)، تنها به دلیل نشان دادن تفاوت‌هایی است که در علم جدید در حال شکل‌گیری، با علم سابق وجود دارد. و به تدریج که دامنه نظریات و کاربردهای علم جدید وسعت یابد، قطعاً کلیه زمینه‌های علم آموزش در علم آموزش الکترونیک پوشش داده خواهد شد و علم آموزش الکترونیک به علم آموزش تغییر نام خواهد داد. منتهی این علم آموزش، نه آن علمی خواهد بود که در کتاب‌های آن، فصلی برای آموزش الکترونیک اختصاص داده شود. بلکه آموزش الکترونیک در همه جای آن علم رسوخ کرده و حتی به جرات بتوان گفت که به عنوان مبنای اجرائی، کاربردی و عملیاتی آن علم محسوب می‌شود. به عبارت دیگر نظریات مطرح در علم جدید به تمام ابعاد علم اولیه داخل شده و آن‌ها را تحت تاثیر قرار می‌دهد [مجیدی ۸۰]. شهر الکترونیک نیز نوعی از شهر است با خصوصیات و ابعاد متفاوت، و سازوکارهای خدمات شهری متفاوت.

و البته اثر نوع برداشت و ادراک ما از فاوا، در برنامه‌ریزی توسعه در

واقع می‌شود و در این جامعه فناوری‌های پیشرفته برای بهبود شرایط زندگی و کاری برای همه شهروندان به کار گرفته می‌شود. [Hafkin 2004]

- از دیدگاه فرانک وبستر، پنج تعریف از جامعه اطلاعاتی را که هر کدام معیاری برای تشخیص جامعه نو عرضه می‌کنند می‌توان در قلمرو تحلیل این پدیده در نظر گرفت. از منظر فناورانه، جامعه اطلاعاتی، بر نوآوری فوق‌العاده فناورانه بنا شده است و پیشرفت‌های خیره‌کننده در پردازش، نگهداری، و انتقال اطلاعات به کاربرد فناوری‌های اطلاعاتی در تمامی زوایای حیات اجتماعی انجامیده است. از منظر اقتصادی، تغییر وضعیتی از اقتصاد کالایی به اقتصاد اطلاعاتی انجام شده است و جامعه‌ای خلق شده است که مشخصه بارز آن این است که دانش و سازمان، آفرینندگان اصلی ثروت می‌باشند. از منظر شغلی، از هنگامی که مشاغل اطلاعاتی به وجه غالب در میان مشاغل موجود تبدیل می‌شوند، ما به یک جامعه اطلاعاتی دست می‌یابیم. از منظر مکانی، تاکید عمده بر روی شبکه‌های اطلاعاتی‌ای است که مکان‌های جغرافیایی را به یکدیگر پیوند می‌دهد و در نتیجه آثار شگفت‌انگیزی بر سازمان زمان و مکان دارند. از منظر فرهنگی، فرهنگ کنونی به طور آشکار نسبت به تمامی دوره‌های پیشین سرشارتر از اطلاعات است و ما در محیطی لبریز از رسانه‌ها زندگی می‌کنیم که اطلاعات به درون خصوصی‌ترین قلمرو زندگی مان رسوخ پیدا کرده است. [وبستر ۸۰]
- در تعاریف ارائه‌شده از سوی صاحب‌نظران، پدیده اطلاعات و فاوا، به عنوان مولفه اصلی جامعه اطلاعاتی بیان می‌شود. جامعه اطلاعاتی، نوع خاصی از جامعه است که در آن سیستم‌های رایانه‌ای و فناوری اطلاعات، به طور وسیع به کار گرفته شده است؛ اشکال خاصی از ابعاد کلیدی ۱۲ گانه جامعه را محقق می‌نماید؛ و ابزارها، نظام‌های اجتماعی، فرهنگ، وجوه مشترک و نطائر آن را در طیف خاصی بنا کرده که این طیف، نسبت به انواع جامعه غیر بهره‌مند از این سیستم‌ها و فناوری، متفاوت است.
- تحولات سریع، تنوع و پیچیدگی، از مشخصه‌های کلیدی محیط جامعه اطلاعاتی محسوب می‌شوند. هر ترسیم عقلانی برای یک جامعه اطلاعاتی و مشتقات آن (از جمله شهر الکترونیکی)، نیازمند محسوب نمودن این سه مشخصه کلیدی در معادلات آن ترسیم است. به عبارت دیگر، در صورتی که یک تحلیل، یا طراحی معماری یا ارائه برنامه عملیاتی، بدون ملاحظه و توجه جدی به این سه مشخصه صورت گیرد، محکوم به شکست خواهد بود.
- یک جامعه اطلاعاتی، ما را با طیف متفاوتی از خصوصیات مواجه می‌سازد. برخی از این خصوصیات کلیدی که طیف تفاوت‌های این جامعه را با انواع دیگر آن نشان می‌دهند عبارتند از:
- فراهم بودن زمینه مشارکت بیشتر اقشار مختلف در فعالیت‌های جامعه و حکومت،
- تعاملی بودن بیشتر،

پهنه فناوری اطلاعات نیز متفاوت خواهد بود. در برنامه‌ریزی توسعه یک نظام متأثر از به کارگیری سیستم‌های رایانه‌ای، نوع ادراک ما از لایه‌های تحت تأثیر این سیستم‌ها، در نوع برنامه‌ریزی ما موثر خواهد بود. در صورتی که ادراک ما از تحولات سازمانی، محدود به دولا به اول و دوم باشد، برنامه‌ریزی توسعه ناشی از این سطح ادراک نیز، تنها در خرید و به کارگیری تجهیزات و حداکثر ایجاد برخی زیرساخت‌ها، محدود می‌شود. محدودیتی که متأسفانه بسیاری از حرکت‌های برنامه‌ریزی توسعه در سال‌های اخیر، به آن گرفتار بوده‌اند. در صورتی که ادراک ما از این اثرات، به لایه سوم و ساختار سیستم‌ها کشیده شود، نوع برنامه‌ریزی توسعه ما نیز به اصلاح ساختارها و سیستم‌ها، در پهنه این به کارگیری کشیده خواهد شد. اما زمانی که لایه‌های چهارم و پنجم نیز در دامنه ادراک ما قرار گیرد، برنامه‌ریزی توسعه ما، شکل متفاوتی را خواهد یافت. برنامه‌ریزی توسعه در این حالت به شکل اصلاح ساختارها، بسترهای اجتماعی و محیطی، و قوانین، مبتنی بر نظریات علمی ظاهر می‌شود. وقتی بدانیم که مفاهیم جدیدی بر اثر به کارگیری وسیع و عمیق فناوری اطلاعات در شهر و ارائه خدمات شهری در حال شکل‌گیری است، برای تبیین توسعه مطلوب ناشی از این به کارگیری، باید به تبیین نظری ابعاد مختلف آن پرداخت؛ و در این تبیین، شکل‌گیری نوع جدیدی از شهر و ارائه خدمات شهری را، در قالب پدیده شهر الکترونیکی، دنبال نمود. با چنین نگاهی به فناوری اطلاعات و ارتباطات، و آگاهی از لایه‌های اثرات آن بر سازمان و ساختار شهر و نظام مدیریت شهری، می‌توان به نحو مطلوب‌تری زاویه نگاه خود را در پدیده‌شناسی شهر الکترونیکی، تبیین نمود. [مجیدی ۸۳].

۵- جامعه اطلاعاتی

جامعه اطلاعاتی پدیده‌ای است که تعریف واحد و مورد اتفاق صاحب‌نظران بر آن وجود ندارد. از جمله می‌توان به تعاریف زیر اشاره نمود:

- جامعه اطلاعاتی، جامعه‌ای است که در آن هر فرد می‌تواند نیازهای اطلاعاتی و ارتباطی خود را در آن برآورده سازد، بدون آن که خدشه‌ای به حقوق دموکراتیک دیگران وارد کند. [Gohring 2009]
- جامعه اطلاعاتی، جامعه‌ای است که در آن فاوا امکانات جدیدی را برای ایجاد، توزیع، به کارگیری، و دستکاری اطلاعات در فعالیت‌های اقتصادی سیاسی و فرهنگی گشوده است. [Malinauskiene 2009]
- جامعه اطلاعاتی، جامعه‌ای است که در آن افراد با فناوری به عنوان بخش مهمی از زندگی و تشکیلات اجتماعی تعامل می‌کنند تا بتوانند اطلاعات را در سطح جهانی مبادله کنند. این نوع جامعه از طریق تغییرات ایجاد شده در حوزه‌های مختلف فاوا مورد تأثیر

۶- جامعه تمدنی و شهر

شهر، اولاً یک جامعه تمدنی است. جامعه تمدنی، جامعه‌ای است که همانند تمامی جوامع، روح و فضای تصمیم‌گیری و نیز رفتار یکپارچه‌ای دارد، اما این نوع جامعه، رفتار یک جامعه متمدن را از خود بروز می‌دهد. لزوماً هر جامعه‌ای رفتار متمدنانه‌ای ندارد، و جوامع ابتدائی و جوامع وحشی جزء جامعه متمدن محسوب نمی‌شوند، هر چند که در آن‌ها نیز قوانین و مقرراتی حاکم است. برخی از خصوصیات یک جامعه متمدن، دارای اهمیت بیشتری در پدیده‌شناسی شهر الکترونیکی است. از جمله این خصوصیات عبارتند از:

- وجود، شناسائی و رعایت نسبی حریم‌ها؛ در یک جامعه متمدن، حریم‌های خاصی توسط نظام تمدنی مربوطه تعیین، و توسط افراد جامعه رعایت می‌شوند.
- ادراک ابعاد مسائل اجتماعی مبتلا به توسط جامعه؛ سطحی از عقلانیت در یک جامعه متمدن شکل می‌گیرد که رفتارهای اجتماعی و فضاهای تصمیم‌گیری اجتماعی، مبتنی بر این عقلانیت بنا می‌شود.
- ایجاد شکل خاصی از نظم در تعاملات اجتماعی (نظم تمدنی)؛ قوانین و مقررات در یک جامعه متمدن به وضوح احساس می‌شود و افراد در شرایط مشخص، رفتار کاملاً مشخصی را از خود بروز می‌دهند. در این جامعه، از ابزارهای خاصی برای به کارگیری نظم استفاده می‌شود.

تمدن یک جامعه تمدنی، نسبی است. ممکن است جامعه‌ای از جامعه دیگر متمدن‌تر باشد و یا این که جوامع متمدن مختلف در یک حوزه مشخص مانند ارائه خدمات و رفاه، رفتارهای متفاوتی داشته باشند. در یک جامعه تمدنی نیز، قواعد و مقررات و فرهنگی وجود دارد که در جامعه نهادینه شده است. نکته مهم آن است که سه خصوصیت ذکر شده، در هر سطح از پیشرفتگی تمدنی یک جامعه متفاوت خواهد بود. به عبارت دیگر یکی از معیارهای شناسائی پیشرفتگی تمدنی هر جامعه، شناسائی این سه خصوصیت و پیشرفتگی آن خواهد بود. شهر الکترونیکی نیز به عنوان یکی از سطوح جامعه تمدنی، از این موضوع مستثنی نیست. نوع خاصی از حریم‌ها و شناسائی آن، ادراکات و عقلانیت، و نظم در شهر الکترونیکی وجود دارد، که شاخصه تمدنی آن شناخته می‌شود.

در یک تعریف از پدیده "شهر":

"شهر زیستگاهی است انسان ساخت و در زیر یک قدرت سیاسی مشخص، که تمرکز جمعیتی نسبتاً پایداری را درون خود جای می‌دهد، فضاهائی ویژه بر اساس تخصص‌های حرفه‌ای به وجود می‌آورد، تفکیکی کم و بیش مشخص و فزاینده میان بافت‌های مسکونی و بافت‌های کاری ایجاد می‌کند و فرهنگی خاص را به مثابه حاصلی از روابط درونی خویش پدید می‌آورد که درون خود خرده فرهنگ‌های بی‌شماری را حمل می‌کند" [فکوهی ۸۳].

- خارج بودن از محدوده مرزهای رسمی،
 - جهانی‌بودن و متداخل بودن فضاهای تصمیم‌گیری جوامع، و اثرگذاری‌های متقابل در این فضاها،
 - مردم محوری حکومت‌ها به دلیل تعاملی بودن آن،
 - توزیع‌شدگی ساختارها، تصمیم‌ها و رفتارها در بخش‌های مختلف جامعه، در عین یکپارچگی آن‌ها،
 - ویژه‌گرایی و امکان پاسخگویی به نیازهای متنوع افراد،
 - اشکال جدید پدیده‌ها و مسائل اجتماعی، نظام‌ها و ساختارهای اجتماعی، مفهوم توسعه و پیشرفتگی و معیارهای آن،
 - اشتراک وسیع و به جریان افتادن اطلاعات و دانش و تجربیات و علوم،
 - تضادهای (پارادوکس‌ها) نو، از جمله در رشد قدرت شرکت‌های کوچک و سیطره شرکت‌های جهانی بسیار بزرگ، بومی و محلی‌سازی در کنار جهانی‌سازی، و توزیع‌شدگی در کنار تمرکز،
 - توسعه قواعد، سازوکارها و استانداردهای جدید،
 - خودسازمانده و فعال در شکل‌گیری، به صورتی که جامعه دائماً در حال تغییر و سازماندهی مجدد از درون است.
- جامعه اطلاعاتی، می‌تواند منجر به بهبود کیفیت زندگی، کاهش فقر، توسعه، به کارگیری، به جریان انداختن و رشد دادن استعدادهای نهفته و خلاقیت، توسعه دانائی محور، ارتقای آموزش، دانش و مهارت، به جریان انداختن و اشتراک اطلاعات، دانش و تجربیات و علوم، تسهیل و ارتقای تصمیم‌گیری و فرایند آن به شکل توزیع‌شده و مشارکتی شود.
- اما در عین حال عوارض ناخواسته‌ای (نامطلوب یا مطلوب) نظیر شکاف دیجیتالی، تعامل بین‌المللی فرهنگ‌ها و زبان‌ها، برداشته شدن مرزها، تغییر وجوه حاکمیت و اقتدار سیاسی، بی‌هویتی فرهنگی از طریق سرایت و تهاجم بی‌ریشه فرهنگی - فرهنگ کاذب، نیز دامنگیر جامعه اطلاعاتی است که باید مورد توجه قرار گیرند و تلاش برای کاهش یا تعامل مطلوب با آن صورت گیرد.
- اما تحقق جامعه اطلاعاتی مستلزم فراهم آمدن ملزوماتی است. غیرفنی‌سازی در حوزه کاربری فاوا، فرصت‌های عادلانه و ملزومات در استفاده از فاوا، فرصت‌های ویژه و یارانه‌ای در کاربرد فاوا، خارج شدن از حصار و ابزارهای دولت‌ها، مشارکت مردم - حکومت در شکل دولت، ایجاد چارچوب‌های قانونی و سیاسی جدید و متفاوت، احترام به آزادی‌های معقول و فراهم آوردن فرصت برای ارائه دیدگاه‌ها و نظرات و مشارکت در فعالیت‌ها، جریان آزاد ایده‌ها، حفظ حریم‌ها و اطمینان و امنیت و ایمنی، ارتقای دانش، مهارت، فرهنگ و آگاهی عمومی نیروی انسانی، تعامل و مشارکت جهانی، درک فرهنگ‌های متفاوت و تعامل فرهنگی آگاهانه، چندزبانی شدن بسترهای اطلاعاتی، و رشد عقلانیت اجتماعی، از جمله ملزومات تحقق یک جامعه اطلاعاتی، محسوب می‌شوند. [مجیدی ۸۵-۱]

صورت تخصصی در حوزه‌های خدماتی مورد پاسخگویی واقع شود.

- نوع تقسیم فضا و شکل‌گیری کالبدی، سازوکارهای خاصی را برای مدیریت فضا و مدیریت فعالیت‌های مرتبط با فضا ایجاد می‌کند، که باید در شهر به آن‌ها پاسخ داده شود.

۷- شهر اطلاعاتی یا شهر الکترونیکی

حال اگر بین تعریف و خصوصیات شهر، و تعریف و خصوصیات جامعه اطلاعاتی و در پس آن مفهوم جامعه، فناوری اطلاعات و رایا سپهر تعامل مناسبی برقرار شود، خصوصیتی را می‌توان برای شهر اطلاعاتی یا شهر الکترونیکی بر شمرده که به ما در تحلیل رویکرد تحول و تحقق شهر الکترونیکی کمک خواهد نمود.

- شهر الکترونیکی علاوه بر آن که زیستگاهی فیزیکی را برای زندگی شهروندان فراهم می‌کند، زیستگاهی مجازی را نیز به موازات آن برای فعالیت‌ها و تعاملات آن‌ها ایجاد می‌کند. حتی امکان بسیاری از نیازهای روزمره افراد باید از طریق این زیستگاه و فضای مجازی برآورده شود.

- با توجه به وجه زیستگاه مجازی شهر الکترونیکی، شهروندانی نیز می‌توانند به عضویت یک شهر درآیند که به صورت فیزیکی عضو و ساکن آن شهر نیستند. در بستر خدمات الکترونیکی ارائه خدمات به شهروندان مجازی نیز معنا پیدا می‌کند. شهروندانی که در نقطه دیگری از کشور یا دنیا زندگی می‌کنند. این تعامل ممکن است در نقش ارتباط شخص با زادگاه خود، ارتباط با محیط زندگی اقوام، ارتباط با محیط کسب و کار، ارتباط با محیط‌های آموزشی، ارتباط گردشگری و انواع متعدد دیگری از ارتباط باشد.

- در شهر الکترونیکی، تنوع فعالیت‌ها افزایش چشم‌گیری یافته و چارچوب‌های قبلی برای فعالیت‌ها، پاسخگوی نیازهای این تنوع در شکل و ماهیت فعالیت‌ها نخواهد بود. به همین لحاظ چارچوب‌های فعالیت‌ها دچار تغییرات اساسی می‌گردد.

- همان‌طور که بازار در شهر نقش محوری را ایفا می‌کند، در شهر الکترونیکی هم بازار منتهی با شکل دیگری از آن نقش بسیار کلیدی را دارا است. تنوع فعالیت‌ها و شکل ارائه خدمات در فضای مجازی، برای رشد نیازمند پشتیبانی سازوکار مناسب بازار در این فضا است. بسیاری از آنچه که در شهرهای غیرالکترونیکی به شکل جدی در محدوده فعالیت بازار وارد نشده‌اند، در این فضا در محدوده بازار وارد می‌شوند.

- در شهر الکترونیکی، نهادهای مدنی مجازی با توجه به سادگی در تاسیس و فعالیت آن‌ها، رشد روزافزونی می‌یابند و انواع جدیدی از نهادهای مدنی ایجاد می‌شوند.

- در شهر الکترونیکی، شکل‌های متفاوتی از جریان قدرت ظهور پیدا

از منظر پدیده‌شناسی شهر الکترونیکی، چند خصوصیت از پدیده شهر قابل توجه به نظر می‌رسد [فکوهی ۸۳] [و بر ۷۵] (و دیدگاه‌های نویسنده):

- شهر یک زیستگاه است. انسان‌ها در آن زندگی و فعالیت می‌کنند. شهر و بسترهای خدمات شهری باید بتواند نیازهای روزمره انسان‌ها را برآورده سازند.
- شهر دارای موقعیت جغرافیائی است؛ و در آن متمرکز است. مولفه‌های تشکیل‌دهنده یک شهر، دارای نوعی از همجواری هستند، و تعاملات مکانی و فیزیکی بین آن‌ها از این منظر دارای اهمیت ویژه‌ای است.
- در یک شهر تنوع زیادی در انجام فعالیت‌ها و معاملاتی که در آن صورت می‌گیرد، وجود دارد.
- علیرغم تنوع زیاد مذکور، فعالیت‌ها، مشاغل و کسب و کار در شهر، دارای چارچوب‌های مشخص و کلیشه‌شده‌ای است که فعالیت یک نفر الزاما باید از طریق آن‌ها صورت گیرد.
- بازار در زندگی شهری، نقش حیاتی را ایفا می‌کند. برخلاف روستاها که نقش بازار در حیات یک مجموعه روستائی آن‌چنان پیچیده نیست (از تعداد مغازه‌های یک روستا می‌توان به این موضوع پی برد) و تعاملات مالی و کالا به صورت مستقیم بین افراد پاسخگوی بسیاری از نیازمندی‌های آنان است، در شهر اغلب موارد نیاز افراد توسط بازار و تعامل مردم با این پدیده صورت می‌گیرد.
- نهادهای خاص مدنی، و سازماندهی، فعالیت و خدمات آن‌ها، نقش محوری را در حیات یک شهر ایفا می‌کنند.
- فرهنگ و روابط انسانی خاصی در شهر ایجاد می‌شود، که هر گونه فعالیت و نظام عملکردی، باید متناسب با آن صورت گیرد.
- سیاست و جریان قدرت در شهر، سازوکار مشخصی دارد که فعالیت تمام مولفه‌های شهری، باید متناسب و در تعامل با آن صورت‌بندی و سازمان یابد. ضمن آن که این سازوکار، در بسیاری از موارد بستری را برای رفتارهای یکپارچه سیاسی ایجاد می‌کند.
- در شهر ضمن افزایش یافتن فردیت و شخصیت فردی و استقلال رفتار فرد از خانواده (نسبت به روستا که رفتارها عمدتاً در بستر خانوادگی شکل می‌گیرد) از یک سو، در سوی دیگر رفتار فرد با قرار گرفتن در بستر منسجم رفتارهای اجتماعی، در رفتار جمعی بزرگ حل می‌شود. به عبارت دیگر نقش متعادل خانواده در رفتارهای اجتماعی، جای خود را به رفتارهای فردگرا شده از یک سو و جماعت‌گرا شده از سوی دیگر می‌دهد.
- تقسیم کار اجتماعی در شهر به شدت افزایش یافته، و دامنه بسیاری از نیازمندی‌های هر فرد که در روستا توسط افراد خانواده پاسخ داده می‌شد، به بیرون خانه کشیده شده، باید به

می‌کند، که البته سازوکارهای خاصی را نیز برای مدیریت آن طلب می‌کند.

• در شهر الکترونیکی، سوق به سمت رفتارهای فردگرا شده، به شدت تشدید می‌شود. ضمن آن که بروز رفتارهای جماعت‌گرا شده، از حوزه عینی خارج و بیشتر به حوزه ذهنی و فرهنگ وارد می‌شود.

• شهر الکترونیکی در بر دارنده تقسیم کار متفاوت و خاص خود خواهد بود. ضمن این که در برخی از حوزه‌ها تقسیم کار تشدید می‌شود، در برخی از حوزه‌ها تلاقی و یکپارچگی کارها و مشاغل صورت می‌گیرد.

• پایداری سازه‌های فضای مجازی شهر الکترونیکی، بسیار کوتاه‌تر از پایداری فضای فیزیکی آن خواهد بود و دستخوش تغییرات دائمی و بعضاً ناگهانی می‌شود. [مجیدی ۸۶]

۸- خصوصیات رویکرد تحول و تحقق شهر الکترونیکی

بر اساس تحلیل اجمالی که از ابعاد پدیده شهر الکترونیکی صورت گرفت، می‌توان به صورت مختصر محورهای خصوصیات رویکرد تحول شهر الکترونیکی را این چنین بر شمرد:

• با توجه به ماهیت جامعه و فرهنگ در پدیده شهر الکترونیکی، طرح‌های توسعه بسترهای خدمات الکترونیکی نه تنها به شکل ایجاد ابزارها، بلکه به صورت مجموعه‌ای از اقدامات برای شکل‌گیری فرهنگ (و رفتارها و فرسازه‌های مربوط به بستر مذکور)، نهادسازی، ایجاد بسترهای قانونی، ایجاد سازوکارها و سامانه‌های غیرالکترونیکی، و نیز ایجاد بسترهای فنی و سامانه‌های الکترونیکی سامان یابند.

• با توجه به همان، طرح‌های تحول باید به صورتی سامان یابد که انگیزه و کشش لازم را در جامعه برای حرکت به سمت تحول ایجاد کند. جامعه خود باید تحول مورد نیاز را طلب کند. طرح‌های تحول باید درونی (در جامعه) شود. طرح‌های تحول باید دارای جهت‌گیری‌های منطقی و متناسب با منطق جامعه (با قدرت ارتقاء منطق)، سازوکارهای موثر، و قوای محرکه کافی برای ایجاد حرکت در جامعه، نه در تبعیت از طرح، بلکه در تحریک و ایجاد کشش در جامعه برای طلب نمودن اهداف طرح باشد. اقدامات اجرایی حتی‌الامکان بر عهده خود جامعه شهر گذاشته شود.

• با توجه به همان، امکان سنجی طرح‌های تحول باید با در نظر گرفتن قابلیت‌های فرهنگی انجام شود.

• با توجه به همان نکته قبلی، طراحی معماری نظام‌ها باید متناسب با رفتارها و فرسازه‌های موجود و قابل ایجاد صورت گیرد. در طراحی معماری، مشخصاً باید بستر و فضای تصمیم‌گیری

جامعه ترسیم شده، و مبتنی بر آن مدل‌های معماری ارائه گردد. • با توجه به همان، جنس معماری باید از معماری سیستم‌ها و مولفه‌های سازمانی- ابزاری سرد، به معماری رفتارها، فرسازه‌ها، موجودیت‌های اجتماعی و عناصر یک موجود زنده (جامعه) تبدیل شود.

• با توجه به نقش سازمان‌های غیررسمی در جامعه تشکیل دهنده شهر، نقش سازمان‌های غیررسمی و عملکرد آن‌ها در طراحی سیستم‌ها و بسترهای خدمات الکترونیکی مورد توجه قرار گیرد. همچنین مشارکت اقشار مختلف جامعه در طراحی، ایجاد، بهره‌برداری و نگهداری زیرساخت‌های خدماتی مد نظر قرار گیرد.

• با توجه به این که شهر ماهیتاً یک جامعه است، و جامعه نیز برای جامعه بودن خود نیازمند اصل پایداری است، طرح‌های توسعه شهر الکترونیکی باید هر گونه حرکت و ایجاد تحول را با حفظ وضعیت پایداری جامعه، و به شکل تکامل تدریجی انجام دهد. جامعه با هر رفتار مخدوش‌کننده پایداری آن به صورت ناخودآگاه مقابله نموده، در چنین صورتی یا حرکت تحول دچار توقف و خلل خواهد شد، یا در صورتی که فشار زیادی برای تحول به جامعه وارد شود، و جامعه با زور مجبور به پذیرش تحول گردد، ساخت‌ها و مولفه‌های سازنده جامعه در شکل باثبات آن از هم پاشیده شده، و ساخت‌های نامتوازن و معیوب نظیر ساخت‌های انگلی، بازار سیاه، پدیده‌های زیرزمینی، در جامعه رشد می‌کند و بسیاری از سازوکارهای موثر و موجود جامعه اثربخشی خود را از دست می‌دهد.

• با توجه به نقش خودسازماندهی در جامعه از یک سو، و ماهیت خودسازماندهی رایا سپهر، نوع ساخت‌ها و معماری نظام‌های خدمات شهر الکترونیکی، باید مبتنی بر اصل خودسازماندهی بنا شوند. موجودیت خودسازمانده جامعه شهر الکترونیکی نمی‌تواند به وسیله نظام‌های الکترونیکی و غیرالکترونیکی غیر خودسازمانده سازمان یابد. ضمن آن که با توجه به خصوصیات محیطی جامعه اطلاعاتی، برای تعامل صحیح نظام‌ها با تحولات سریع محیط، دستیابی به خودسازمانی نظام‌ها ضرورت بیشتری می‌یابد.

• با توجه به مفهوم فناوری اطلاعات و لایه‌های پنج‌گانه تحول سازمان، طرح‌های توسعه شهر الکترونیکی باید با تحلیل مناسب ماهیت تحول و سازه‌های در حال شکل‌گیری، و اتخاذ راهکارها، راهبردها و رویکردهای متناسب با آن انجام شود.

• مبتنی بر خصوصیات و ملزومات جامعه اطلاعاتی، در ایجاد بسترهای خدمات شهر الکترونیکی باید نقش مردم پرنرنگ‌تر از گذشته ترسیم شود. بسترها حتی‌الامکان توسط بخش خصوصی سازمان یابد. امکان ایجاد و ارائه خدمت بسترها توسط افراد مختلف و حتی به صورت موازی وجود داشته باشد.

الکترونیکی باید نقش خانواده و اهمیت آن را مد نظر قرار داده، و حتی الامکان با اتخاذ تدابیر مناسب، به تشدید واگرایی به سوی فردگرایی و جماعت‌گرایی دامن نزند. این مهم از جمله با در نظر گرفتن نقش‌های مشخص برای خانواده‌ها و تلقی خانواده به عنوان موجودیت‌های اجتماعی قابل شناسایی در فضای الکترونیکی، قابل تحقق است.

• مبتنی بر همان و نیز ماهیت جامعه و عوارض جامعه اطلاعاتی، رعایت ملاحظات فرهنگی در تحقق شهر الکترونیکی، از وجه عوارض فرهنگی ناخواسته و فرهنگ ناموزون باید مورد توجه قرار گیرد.

• مبتنی بر همان، سازوکارهای لازم برای ایجاد و رفتارهای نهادهای مدنی، و جریان قدرت و فعالیت‌های سیاسی، متناسب با نیازمندی‌های شهر الکترونیکی باید ایجاد شود.

• مبتنی بر همان، توجه به سازوکارهای تعامل مناسب با مدیریت فضا، و نیز مدیریت مشاغل، در بسترهای شهر الکترونیکی ضروری است.

• با توجه به خصوصیات شهر الکترونیکی، باید بسترهای مختلف تعامل با شهروندان مجازی (غیر ساکن) در بسترهای خدمات، سازوکارها، قوانین و سازماندهی شهر الکترونیکی پیش‌بینی شود.

• مبتنی بر همان، برای پاسخگوئی به تنوع و نیز تفاوت و تنوع چارچوب‌ها، چارچوب‌های فعالیت‌ها باید در یک بازطراحی بنیادین و مبتنی بر تحلیل نظری پدیده در حال شکل‌گیری، شرایط فعالیت‌ها را به صورت متناسب با نیازمندی‌های شهر الکترونیکی فراهم کند. بخصوص در بسیاری از اقداماتی که در حال حاضر برای ارائه خدمات الکترونیکی (با تبدیل کردن همان فرم‌ها و قالب‌های فعالیت‌های غیرالکترونیکی در محیط مجازی) متداول است، این موضوع قابل توجه به نظر می‌رسد.

• با توجه به همان، و همان نیازمندی به تنوع و تفاوت چارچوب‌ها، بسترهای قانونی و سازوکارهای اجرائی و ضمانت‌های اجرائی متفاوتی مورد نیاز است که باید در طرح‌های تحول به آن‌ها توجه لازم مبذول شود.

• مبتنی بر همان، نقش نظارت و سازماندهی نهادهای مدنی در شهر الکترونیکی دارای اهمیت خاصی خواهد بود. این کار با توجه به دشواری آن در رایا سپهر، نیازمند تمهیدات خاص و اتخاذ نوع متفاوتی از تدابیر در نظارت بر فعالیت نهادهای مدنی خواهد بود.

۹- نتیجه‌گیری

شهر الکترونیکی، از یک سو یک جامعه است، آن هم از نوع جامعه اطلاعاتی، و از سوی دیگر یک شهر است. بنابراین شناخت

• مبتنی بر همان، توزیع‌شدگی سیستم‌ها در اغلب موارد یک ضرورت تلقی می‌شود. بسترهای خدماتی باید امکان توزیع‌شدگی اطلاعات، توزیع‌شدگی خدمات، توزیع‌شدگی مدیریت، و حتی توزیع‌شدگی طراحی و توسعه خود بسترها را داشته باشند.

• با توجه به خصوصیت تنوع در خصوصیات محیط جامعه اطلاعاتی، باید امکان پاسخگوئی به تنوع شهروندان و نیازمندی‌های آنان در بسترهای ارائه خدمات شهر الکترونیکی وجود داشته باشد. خدمات باید به شکل ویژه‌گرایی شده به مخاطبان ارائه شود. بسترها باید از انعطاف‌پذیری و سازوکارهای لازم برای این ویژه‌گرایی برخوردار باشند.

• با توجه به پدیده‌های جدید و شکل جدید پدیده‌ها در جامعه اطلاعاتی، در طرح‌های تحول شهر الکترونیکی، باید تحلیل نظری لازم برای شناسایی ابعاد پدیده‌ها صورت گیرد، و سازوکارهای مناسب برای برخورد با این پدیده‌ها در طرح‌های عملیاتی بر اساس تحلیل‌های نظری مذکور ایجاد شود.

• با توجه به ملزومات جامعه اطلاعاتی، ایجاد چارچوب‌های قانونی مناسب برای شکل‌گیری شهر الکترونیکی ضروری است.

• با توجه به همان، در طراحی بسترهای الکترونیکی، تامین سازوکارهای امنیتی لازم به عنوان یکی از ضروریات تلقی شده، و طرح‌های توسعه باید با در نظر گرفتن این سازوکارها ارائه شوند. طراحی نظام‌ها باید به گونه‌ای صورت گیرد که قابلیت اطمینان کافی را از نظر امنیتی و ایمنی فراهم سازد.

• با توجه به خصوصیات پدیده شهر، در شهر الکترونیکی باید بسترهای خدمات الکترونیکی برای پاسخگوئی به نیازهای روزمره مردم وجود داشته باشد.

• با توجه به همان، با توجه به این که بسترهای خدمات الکترونیکی در یک محیط مجازی ارائه می‌شود که استقلال از فضای جغرافیائی در آن متداول است، پاسخگوئی به نیازمندی جغرافیائی و تعاملات مکانی و فیزیکی شهر، نیازمند اعمال نوعی تعامل و تناظر مکانی در این بسترها است، به گونه‌ای که دریافت‌کننده خدمات به راحتی بتواند خدمات خود را متناسب با حوزه جغرافیائی خود تفکیک، جستجو و دریافت کند. سازماندهی، ممیزی و نظارت بر مولفه‌های فضای مجازی در شهر الکترونیکی نیز باید به گونه‌ای مناسب، با توجه به فضای جغرافیائی صورت گیرد.

• با توجه به همان، بسترهای خدمات و سازماندهی مولفه‌های شهر الکترونیکی باید پاسخگوی نیازهای متنوع شهروندان باشد. • با توجه به همان، و با توجه به نقش حیاتی بازار در پدیده شهر، شکل‌گیری بازار الکترونیکی و ابزارهای آن نظیر بانکداری الکترونیکی و پشتیبانی حمل کالا، از محورهای اساسی شهر الکترونیکی محسوب می‌شود.

• مبتنی بر همان، سازوکارهای سازماندهی و ایجاد بسترهای شهر

ماهیت، تعریف؛ اثر در برنامه‌ریزی توسعه"، گزارش کامپیوتر، مرداد و شهریور ۸۳، شماره ۱۵۸.

[مجیدی ۸۶] - مجیدی، اردوان، درخشان، رضا، ۱۳۸۶، یک معماری کلان برای سازماندهی زیرساخت‌های خدمات در شهر الکترونیک، اولین کنفرانس بین‌المللی شهر الکترونیک - ۱ و ۲ اسفند ۸۶، مرکز همایش‌های بین‌المللی برج میلاد.

[مجیدی ۸۵-۱] - مجیدی، اردوان، ۱۳۸۵، "نگاهی اجمالی به تجربه مطالعه، طراحی معماری و تدوین سند دولت الکترونیک"، فصلنامه تحول اداری، سازمان مدیریت و برنامه‌ریزی کشور، شماره ۵۱.

[مجیدی ۸۵-۲] - مجیدی، اردوان، ۱۳۸۵، "چم: یک چارچوب و متدولوژی معماری سازمانی"، دوازدهمین کنفرانس بین‌المللی انجمن کامپیوتر ایران، دانشگاه شهید بهشتی، ۱-۳ اسفند.

[معماری سازمان فرهنگی ۸۶] - سازمان فرهنگی هنری شهرداری تهران، ۱۳۸۶، "مستندات نهایی معماری سازمان".

[سند دولت الکترونیک ۸۴] - نهاد ریاست جمهوری، تیر ماه ۱۳۸۴، «راهنمای ملی دولت الکترونیک»، مرکز فناوری اطلاعات نهاد ریاست جمهوری، اردوان مجیدی، اسفند ۱۳۸۳، «مستندات نهایی فاز صفر از طرح مطالعه، طراحی معماری و تدوین سند ملی دولت الکترونیک».

[معماری فناوری اطلاعات نهاد ریاست جمهوری، ۱۳۸۳ و ۱۳۸۴]، «مستندات معماری کلان و تفصیلی نظام فناوری اطلاعات و ارتباطات نهاد ریاست جمهوری»، مرکز فناوری اطلاعات نهاد ریاست جمهوری.

[فکوهی ۸۳] - فکوهی، ناصر، ۱۳۸۳، "انسان شناسی شهری"، نشر نی.

[اوبر ۷۵] - وبر، ماکس، ترجمه کاویانی، شیوا، ۱۳۷۵، "شهر در گذر زمان - با پیشگفتار تحلیلی مارتینال"، شرکت سهامی انتشار.

[گیدنز ۸۳] - گیدنز، آنتونی، ترجمه منوچهر صبوری، ۱۳۸۳، "جامعه شناسی"، نشر نی.

[مالینوفسکی ۸۳] - مالینوفسکی، برانیسلاو، ترجمه منوچهر فرهومند، ۱۳۸۳، نظریه علمی فرهنگ، دفتر پژوهش‌های فرهنگی.

[ارفع پور ۸۲] - رفیع پور، فرامز، ۱۳۸۲، آناتومی جامعه، شرکت سهامی انتشار. [وبستر ۸۰] - وبستر، فرانک، ترجمه اسماعیل قدیمی، ۱۳۸۰، نظریه‌های جامعه اطلاعاتی، نشر قصیده‌سرا.

[Srinivasan 2004] - Srinivasan, K., 2004, Artificial Societies and Agent Technology, Indian Institute of Management and Technology.

[Howitt 2009] - Howitt, Bernie, Julian, Robin, 2009, society & culture, 2nd ed, Sydney.

[Gohring 2009] - Gohring, Wolf, 2004, The Memorandum "Sustainable Information Society", 18th International Conference "Informatics for Environmental Protection", EnviroInfo 2004, 21-23 October.

[Malinauskiene 2009] - Malinauskiene, Egle, 2009, E-Government as a key driver for Information Society, Mykolas Romeris University.

[Hafkin 2004] - Hafkin, Nancy, 2004, Globalization And The Economic Empowerment of Women: Defining And Building a Gender-Responsive Information Society In The Escap Region, Economic And Social Commission For Asia And The Pacific.

خصوصیات و ویژگی‌های جامعه اطلاعاتی و شهر به ما کمک زیادی را در شناخت پدیده شهر الکترونیک خواهد نمود. بر این اساس می‌توان ضمن شمارش خصوصیات شهر الکترونیک، خصوصیات رویکرد تحول شهر الکترونیک را نیز برشمرد. در یک جمع‌بندی با توجه به خصوصیات شمارش شده رویکرد تحول و تحقق شهر الکترونیک، می‌توان دو دیدگاه مختلف را به پدیده شهر الکترونیک و تحول آن ترسیم نمود. انتخاب یک دیدگاه، نوع برداشت و واکنش‌های ما را در برنامه‌ریزی توسعه، نسبت به این پدیده، تعیین می‌کند. در دیدگاه اول که بسیار متداول و شایع است، پدیده شهر الکترونیک، تعدادی سیستم یا وبگاه اینترنتی است که اطلاعات و خدمات سازمان‌ها و بنگاه‌ها را به مشتریان ارائه می‌کند. اما در دیدگاه دوم، شهر الکترونیک نوعی تحول یافته از شهر است، که خصوصیات، ملزومات و سازوکارهای خاصی را طلب می‌کند. شکل‌گیری پدیده شهر الکترونیک، شکل‌گیری نوع جدیدی از شهر، و در بردارنده نوع جدیدی از جامعه است، و تنها به کارگیری برخی ابزارها نیست. طرح‌های توسعه شهر الکترونیک، باید بتواند در تجسم چنین پدیده‌ای، با ترسیم و توصیف بنیان‌های نظری، معماری مبتنی بر این بنیان‌ها، و استانداردها و ضوابط تحقق این معماری، سطحی از بسترهای خدمات شهری را ایجاد کند.

قدردانی

نتایج این مقاله با عنایت و یاری خداوند متعال با توجه به نتایج چند طرح و پژوهش انجام شده قبلی حاصل شده است. لازم است از زحمات تمام همکارانی که در انجام مطالعات مربوطه به انحاء مختلف نقش داشته‌اند، تشکر شود. بخصوص همکاران و دست‌اندرکاران مطالعه سند دولت الکترونیک کشور، طراحی معماری سازمان فرهنگی هنری شهرداری تهران، و طراحی معماری نظام امنیت زیرساخت‌های حیاتی کشور، و اختصاصاً توجه و یاری آقای مهندس سید هادی سجادی، آقای مهندس مجتبی جعفری، آقای مهندس حمید رضا رستمی، آقای امیر مسعود حبیبیان، و برخی از همکاران مرکز فناوری اطلاعات نهاد ریاست جمهوری، مرکز تحقیقات مخابرات ایران، سازمان فرهنگی هنری شهرداری تهران، و موسسه مطالعات راهبردی فناوری اطلاعات که ذکر نام همه آن‌ها در این متن مقدور نیست، تشکر و قدردانی گردد.

منابع

[مجیدی ۸۰] - مجیدی، اردوان، ۱۳۸۰، نظام برتر، آینده آموزش و آموزش آینده، نشر ترمه.
[مجیدی ۸۳] - مجیدی، اردوان، ۱۳۸۳، "فناوری اطلاعات و ارتباطات: مفهوم،

بررسی عملکرد انجمن‌های علمی کشور

جدول شماره ۱، تعداد انجمن‌ها و عملکردهای ارائه شده را به تفکیک گروه‌ها نشان می‌دهد.

نام گروه‌ها	تعداد انجمن‌ها	تعداد عملکرد ارائه شده در سال ۸۹
علوم پایه	۲۳	۲۱
فنی مهندسی	۷۴	۵۷
علوم انسانی	۷۷	۵۶
کشاورزی	۳۷	۲۸
بین‌رشته‌ای	۳۴	۲۶
هنر	۳	۱
	۲۴۸	۱۸۹

۲- رتبه انجمن‌های علمی

کمیسیون انجمن‌های علمی، بر اساس معیارهای در نظر گرفته شده،

یکی از وظایف مهم کمیسیون انجمن‌های علمی وزارت علوم، تحقیقات و فناوری، نظارت و ارزیابی مستمر فعالیت‌های انجمن‌های علمی کشور است. گزارش ارزیابی عملکرد انجمن‌های علمی در خصوص فعالیت‌های سال ۱۳۸۹ آن‌ها از سوی این کمیسیون آماده گردیده که توجه خوانندگان گرامی گزارش کامپیوتر را به نکات برجسته این گزارش جلب می‌کنیم.

۱- تعداد انجمن‌ها و عملکردهای ارائه شده

انجمن‌های علمی کشور در شش گروه علوم پایه، فنی مهندسی، علوم انسانی، کشاورزی، بین‌رشته‌ای و هنر تقسیم‌بندی شده‌اند. ۱۸۹ انجمن علمی نسبت به انعکاس عملکرد خویش به کمیسیون اقدام کرده‌اند و ۵۹ انجمن نیز فاقد عملکرد بوده‌اند. همچنین تعداد ۲۴ انجمن نیز در این سال تازه تأسیس بوده‌اند.

کارگاه، سخنرانی و کنفرانس بین‌المللی، همکاری و عضویت در مجامع بین‌المللی و داشتن کمیته بین‌المللی اشاره کرد. جدول ۴، فعالیت بین‌المللی انجمن‌های علمی را به تفکیک گروه‌ها نشان می‌دهد.

نام گروه‌ها	رتبه A	رتبه B	رتبه C	رتبه D	فایده عملکرد	جدید التاسیس	تعداد انجمن‌ها
علوم پایه	۳	۶	۶	۶	۲	۰	۳۳
فنی مهندسی	۱۹	۹	۱۶	۱۳	۱۷	۷	۷۴
علوم انسانی	۱۵	۶	۱۹	۱۶	۲۱	۱۰	۷۷
کشاورزی	۳	۷	۸	۱۰	۹	۴	۳۷
بین‌رشته‌ای	۶	۳	۴	۱۳	۸	۳	۳۴
هنر	۱				۲	۰	۳
جمع	۲۴۸						

نام گروه‌ها	سایت دو زبانه	مسابقه علمی در سطح بین‌المللی	نمایشگاه بین‌المللی	کارگاه بین‌المللی	سخنرانی بین‌المللی	کنفرانس و عضویت بین‌المللی	همکاری کمیته بین‌المللی	جمع
علوم پایه	۹	۶	-	-	۱	۳	۱۵	۴۳
فنی مهندسی	۱۴	-	۲	۱	۱۵	۱۶	۲۴	۸۴
علوم انسانی	۱۱	-	-	-	۵	۱۶	۱۳	۶۷
کشاورزی	۸	-	-	۱	۲	۴	۹	۲۹
هنر	-	-	-	-	-	-	-	۰
بین‌رشته‌ای	۹	-	۱	-	۳	۱۵	۲۶	۶۵
جمع	۵۱	۶	۳	۲	۲۶	۵۴	۸۷	۲۸۸

انجمن‌های علمی را با توجه به میزان فعالیت‌هایشان، به ترتیب از بالاترین تا پایین‌ترین، در رتبه‌های A, B, C, D و F قرار داده است. جدول ۲، تعداد انجمن‌های علمی با رتبه‌های مختلف را به تفکیک گروه‌ها نشان می‌دهد.

نام گروه‌ها	رتبه A	رتبه B	رتبه C	رتبه D	فایده عملکرد	جدید التاسیس	تعداد انجمن‌ها
علوم پایه	۳	۶	۶	۶	۲	۰	۳۳
فنی مهندسی	۱۹	۹	۱۶	۱۳	۱۷	۷	۷۴
علوم انسانی	۱۵	۶	۱۹	۱۶	۲۱	۱۰	۷۷
کشاورزی	۳	۷	۸	۱۰	۹	۴	۳۷
بین‌رشته‌ای	۶	۳	۴	۱۳	۸	۳	۳۴
هنر	۱				۲	۰	۳
جمع	۲۴۸						

۳- نشریات علمی

یکی از مهمترین تولیدات انجمن‌های علمی، نشریات علمی است. تعداد کل مجلات علمی- پژوهشی و علمی- ترویجی ۱۶۲ مجله بوده که از این تعداد ۴ مجله دارای رتبه ISI هستند و ۲۸ مجله به زبان انگلیسی چاپ می‌شوند. جدول ۳، تعداد مجلات علمی برحسب نوع اعتبار علمی و به تفکیک گروه‌ها را نشان می‌دهد.

نام گروه‌ها	پژوهشی	دارای تأخیر در چاپ	ترویجی	دارای تأخیر در چاپ	تعداد کل
علوم انسانی	۶۴	۱۶	۵	۳	۶۹
فنی و مهندسی	۳۱	۱۸	۱۳	۶	۴۴
کشاورزی	۱۹	۴	۲	۱	۲۱
علوم پایه	۲۱	۷	۶	۲	۲۷
هنر	۱	۱	-	-	۱
جمع کل	۱۳۶	۴۶	۲۶	۱۲	۱۶۲

۵- تعداد اعضاء

مهمترین رکن هر جامعه‌ای، مخاطبان و اعضای آن جامعه محسوب می‌گردند. انجمن‌های علمی ایران در پایان سال ۱۳۸۹ دارای حدود یکصد هزار عضو بوده‌اند اما به نظر می‌رسد تعداد اعضاء فعال و پیگیر بسیار کمتر از این مقدار که توسط خود انجمن‌ها اعلام گردیده، باشد. جدول ۵، تعداد اعضاء حقیقی و حقوقی انجمن‌های علمی را به تفکیک گروه‌ها نشان می‌دهد.

نام گروه‌ها	اعضاء	حقیقی	حقوقی
علوم پایه	۱۵۷۷۳	۱۴۱	
فنی مهندسی	۳۲۹۳۲	۳۹۱۴	
علوم انسانی	۲۸۶۹۸	۹۹۳	
کشاورزی	۱۱۳۴۵	۵۵	
بین‌رشته‌ای	۱۰۴۷۰	۲۸۱	
هنر	۳۳۰	۹	

۴- فعالیت‌های بین‌المللی

مشارکت فراگیر و گسترده در مراودات علمی بین‌المللی دارای اهمیت خاصی است که هم موجب شناسایی انجمن‌ها در سطوح بین‌المللی می‌گردد و هم باعث ارتقاء جامعه علمی کشور می‌گردد. از جمله فعالیت‌های بین‌المللی می‌توان به ایجاد وبگاه دو زبانه، برگزاری مسابقه علمی در سطح بین‌المللی، برگزاری نمایشگاه،

۶- همایش‌ها، سخنرانی‌ها و کتاب‌های علمی

برگزاری همایش‌های علمی، سخنرانی‌ها و دوره‌ها و کارگاه‌های آموزشی از نکات مثبت و ارزنده‌ای است که اهتمام ویژه در افزایش سطح این نشست‌ها سبب ارتقاء انجمن‌های علمی می‌گردد. متأسفانه ارزیابی کمیسیون انجمن‌های علمی نشانگر عدم استقبال پرشور اعضای هیأت علمی و حضور کم‌رنگ ایشان در همایش‌ها و نشست‌ها

۷- ایجاد اتحادیه‌های تخصصی

کمیسیون انجمن‌های علمی در راستای تقویت انجمن‌های علمی نسبت به ترغیب و تشویق انجمن‌ها جهت ایجاد اتحادیه‌های تخصصی اقدام نموده است. این اتحادیه‌ها می‌توانند علاوه بر ایجاد هماهنگی و یکپارچگی، باعث هم‌افزایی علمی و نیز توانمندی انجمن‌ها در مواجهه با دستگاه‌ها و مراجع قانونی گردند.

جدول ۷، اتحادیه‌های در شرف تأسیس و در شرف صدور مجوز را نشان می‌دهد.

اتحادیه‌های در شرف تأسیس	اتحادیه‌های در شرف صدور مجوز
اتحادیه انجمن‌های علوم مدیریت اتحادیه انجمن‌های مهندسی و علم مواد	اتحادیه انجمن‌های گیاهان دارویی اتحادیه انجمن‌های علوم زیستی اتحادیه‌های انجمن‌های برق و کامپیوتر اتحادیه انجمن‌های منابع طبیعی و محیط زیست اتحادیه انجمن‌های انرژی اتحادیه انجمن‌های گیاه پزشکی

و عدم وجود نوآوری در برگزاری همایش‌ها و نیاز محور نبودن بسیاری از مقالات ارائه شده در این نشست‌ها می‌باشد.

جدول ۶، تعداد همایش‌ها، سخنرانی‌ها، کتاب‌های علمی، دوره‌ها و کارگاه‌های آموزشی برگزار شده از سوی انجمن‌های علمی در سال ۱۳۸۹ را به تفکیک گروه‌ها نشان می‌دهد.

نام گروه‌ها	همایش‌ها	سخنرانی‌ها	انتشارات	دوره‌های آموزشی، کارگاه‌های تخصصی	جمع
علوم پایه	۶۷	۵۵۷	۱۴۴	۱۰۹	۸۷۷
فنی مهندسی	۱۰۲	۴۹۵	۵۵۵	۵۷۹	۱۷۳۱
علوم انسانی	۸۶	۳۶۰	۷۰۵	۳۶۱	۱۵۱۲
کشاورزی	۳۰	۱۰۲	۱۰۵	۴۵	۲۸۲
بین‌رشته‌ای	۴۰	۱۵۳	۱۶۸	۱۲۷	۴۸۸
هنر	۳	۶	۴	۲	۱۵
جمع	۳۲۸	۱۶۷۳	۱۶۸۱	۱۲۲۳	۴۹۰۵

منتشر شد!



برای کسب اطلاعات بیشتر و تهیه کتاب
با شماره تلفن‌های زیر تماس حاصل فرمایید
۸۸۸۶۱۴۲۱-۳ (انجمن انفورماتیک ایران)

و یا برای خرید اینترنتی به سایت زیر مراجعه فرمایید
www.chare.ir

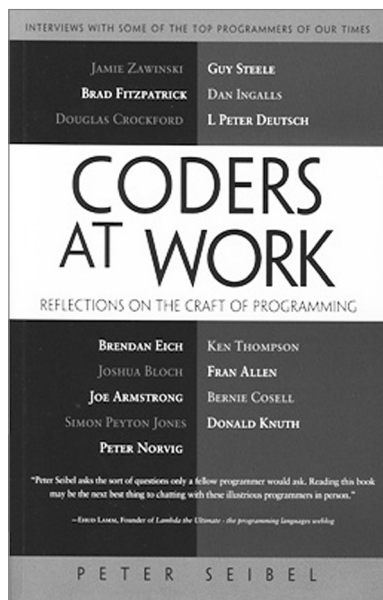
برنامه‌نویسان در حال کار

(قسمت سیزدهم)

کن تامپسون

ترجمه ابراهیم نقیب‌زاده مشایخ

پست الکترونیکی: mashayekh@isi.org.ir



ترجمه این کتاب جذاب و خواندنی، به صورت یک سلسله مقاله دنباله‌دار، به خوانندگان گرامی گزارش کامپیوتر تقدیم می‌گردد. امید است این پاورقی جدید نیز همچون پاورقی‌های قبلی مجله، مورد استقبال خوانندگان قرار گیرد و در انتها در قالب یک کتاب منتشر و در اختیار علاقه‌مندان گذاشته شود.

* * *

یادداشت مترجم

کتاب «Codars at Work» نوشته پتر سیبل در سال ۲۰۰۹ به چاپ رسیده و با وجودی که مدت زیادی از انتشار آن نمی‌گذرد مورد استقبال زیادی از سوی طراحان نرم‌افزار و برنامه‌نویسان قرار گرفته است. پتر سیبل، که نام کتابش را از کتاب‌های معروف و پیشین «Writers at Work» و «Founders at Work» اقتباس کرده، در این کتاب به مصاحبه با پانزده تن از برجسته‌ترین و سرشناس‌ترین برنامه‌نویسان و دانشمندان رایانه، با پیش زمینه تحصیلی و علاقه‌مندی‌های متفاوت در این حوزه، پرداخته و خواننده را با ایده‌های آن‌ها درباره زندگی و برنامه‌نویسی آشنا ساخته است.

اغلب برنامه‌نویسان به تنهایی و یا در گروه‌های کوچک به فعالیت می‌پردازند و جالب‌ترین بخش فعالیتی که انجام می‌دهند در مغز آن‌ها اتفاق می‌افتد که به دور از چشم دیگران است. تنها پنجره‌ای که به کار برنامه‌نویسان وجود دارد، رفتار برنامه‌هایی است که توسط آن‌ها نوشته شده و بر روی رایانه‌ها اجرا می‌شود. بنابراین، اغلب برنامه‌نویسان تنها خودشان، و احتمالاً چند همکار معدودشان، می‌دانند که چگونه برنامه نویسی می‌کنند و چگونه این کار را فرا گرفته‌اند. پتر سیبل به دلیل آن که خود با حرفه برنامه‌نویسی آشنایی کامل دارد، سعی نموده است تا با طرح پرسش‌های جالب و تخصصی، این گوشه‌های پنهان را برای خواننده آشکار سازد.

سیبل: تصوّر محاسبات دودویی را از کجا کسب کرده بودید؟

تامپسون: در کلاس مفهوم مبنای دو را به ما معرفی کرده بودند.

سیبل: آیا شما قربانی «ریاضیات جدید» بودید؟

تامپسون: نه، نه. من قربانی «ریاضیات بد» بودم. ما هر سال محل زندگیمان عوض می‌شد و به این خاطر، من هم به مدرسه‌های خیلی وحشتناک و هم به مدرسه‌های خیلی خوبی رفتم. بنابراین، گاهی مجبور می‌شدم کار دو سال را در یک سال انجام دهم و گاهی یکسال کاری برای انجام دادن نداشتم. در نتیجه، من آموزش ریاضی مناسبی در دوران دبستان نداشتم و وقتم تلف می‌شد. و همان‌طور که گفتم در یکی از این کلاس‌ها محاسبات دودویی را به ما معرفی کردند و من آن را به مبناهای دیگر بسط دادم و سرم را با این چیزها گرم می‌کردم.

سیبل: و این کارها را در دبستان می‌کردید؟!

تامپسون: بله، کلاس هفتم. (در نظام قدیم آموزشی آمریکا، از کودکستان تا کلاس هشتم، دوره آموزش ابتدایی بود و از کلاس نهم تا دوازدهم، دوره آموزش دبیرستانی. مترجم) سپس در دوران دبیرستان، خیلی به الکترونیک علاقه‌مند شدم و رادیو و تقویت‌کننده^{۱۲} و نوسانگر^{۱۳} می‌ساختم. و همچنین شیفته راینش قیاسی شدم. واقعاً شگفت‌آور بود. الکترونیک تنها عشق من در آن دوران بود. من بعداً وارد رشته مهندسی برق در دانشگاه برکلی شدم و در آنجا بود که برای نخستین بار رایانه‌های رقمی واقعی را دیدم.

سیبل: و این چه سالی بود؟

تامپسون: بهار یا پائیز ۱۹۶۲. آن‌ها یک رایانه قیاسی و یک رایانه طبله‌ای^{۱۴} G15 در آزمایشگاه داشتند. تقریباً هیچکس به سراغ رایانه قیاسی نمی‌آمد و من به طور انحصاری با آن کار می‌کردم و لذت می‌بردم. من خودم برنامه‌هایی هم برای مقیاس‌بندی یک رایانه قیاسی نوشتم. همان‌طور که می‌دانید راینش قیاسی تقریباً تماشش مقیاس‌بندی است.

سیبل: مقیاس‌بندی چی؟

تامپسون: مقیاس‌بندی زمان و دامنه^{۱۵}. اساساً کاری که شما می‌کنید این است که آن را برای انجام یک تابع می‌سازید. چند ورودی وارد می‌کنید و تابعی از آن ورودی دریافت می‌کنید و این‌ها را به بازخورد^{۱۶} الحاق می‌کنید. و در هیچ نقطه‌ای از این فرایند نمی‌توانید خیلی بالا بروید. مقیاس‌بندی زمانی هم همین‌طور است. بسامد را در مکان‌های مختلف نصف یا دو برابر می‌کنید. و هنگامی که این کار را می‌کنید، مقیاس‌بندی خطی هم تغییر می‌کند. بنابراین، اگر کار ساده‌ای داشته باشید که به مقیاس‌بندی نیاز نداشته باشد، رایانه قیاسی واقعاً عالی است. اما به محض آن که به مقیاس‌بندی نیاز باشد، بسیار بسیار پیچیده می‌شود. به همین خاطر، همان‌طور که گفتم، برنامه‌هایی

12- amplifier

13- oscillator

14- drum computer

15- time & amplitude scaling

16- feedback

کن تامپسون^۱ تمام دوران زندگی حرفه‌ایش را به انجام کارهایی که برایش جالب بوده گذرانده و این کارها در زمان‌های مختلف، چیزهای مختلفی از قبیل راینش قیاسی^۲، برنامه نویسی سیستم^۳، عبارت‌های منظم^۴ و شطرنج رایانه‌ای بوده است.

تامپسون به عنوان پژوهشگر به استخدام آزمایشگاه‌های بل^۵ درآمد تا بر روی پروژه مالتیکس^۶ کار کند و پس از آن که آزمایشگاه‌های بل خود را از این پروژه کنار کشید، تامپسون به کارش ادامه داد و همراه با دنیس ریچی^۷ سیستم عامل یونیکس را ابداع کرد و جالب آن که کاملاً انتظار داشت به خاطر این کار اخراجش کنند. او همچنین زبان برنامه‌نویسی B را ابداع کرد که پیش درآمدی بود برای زبان C دنیس ریچی.

او بعداً به شطرنج رایانه‌ای علاقه‌مند شد و نخستین رایانه خاص منظوره^۸ برای بازی شطرنج به نام Belle را ساخت که در زمان خود قوی‌ترین شطرنج‌باز رایانه‌ای بود. او همچنین به توسعه دادگان^۹ پایان بازی‌های شطرنج کمک کرد تا تمام پایان بازی‌های چهار و پنج مهره‌ای را در بر گیرد.

تامپسون هنگامی که در آزمایشگاه‌های بل بر روی سیستم عامل Plan9 کار می‌کرد، سیستم کدگذاری UTF-8 را ابداع کرد که همچنان به‌طور گسترده‌ای مورد استفاده است.

تامپسون و ریچی در سال ۱۹۸۳ به خاطر «توسعه نظریه سیستم‌های عامل و به ویژه، پیاده‌سازی سیستم عامل یونیکس» برنده جایزه تورینگ^{۱۰} شدند. او همچنین به خاطر کارهایش بر روی سیستم عامل یونیکس، مدال ملی فناوری و نیز جایزه تسوتومو کانائی^{۱۱} انجمن مهندسان برق و الکترونیک (IEEE) را دریافت کرده است.

من در این مصاحبه با او درباره عشق و علاقه اولیه‌اش به علم الکترونیک، دوران حرفه‌ای نامتقارنش و نیز این که چرا از برنامه‌نویسی مدرن هراس دارد، گفتگو کرده‌ام.

* * *

سیبل: برنامه‌نویسی را چگونه آموختید؟

تامپسون: من همیشه شیفته منطق بودم و حتی در دبستان بر روی مسایل محاسباتی در مبنای دو و چیزهایی از این قبیل کار می‌کردم. واقعاً شیفته این کارها بودم. یادم می‌آید که الگوریتمی برای جمع در مبناهای مختلف نوشته بودم.

1- Ken Thompson

2- analog computing

3- system programming

4- regular expressions

5- Bell Labs

6- MULTICS

7- Dennis Ritchie

8- Special-purpose

9- database

10- Turing Award

11- Tsutomu kanai

بر روی رایانه رقمی برای مقیاس‌بندی برپاسازی‌های^{۱۷} رایانه قیاسی نوشتیم. بدون محاسبه شکل موج واقعی، شما دامنه و بسامد شکل موج را در هر نقطه محاسبه می‌کنید. و به شما می‌گوید که چه موقع از محدوده خارج شده‌اید.

سیپیل: برنامه‌های شما بر روی رایانه رقمی به زبان همگذاری (اسمبلی) بود یا فرتن؟

تامپسون: عمدتاً به زبان همگذاری (اسمبلی) بود. یک زبان تفسیری^{۱۸} بود که خیلی هم کند بود. به همین دلیل من مجبور شدم به سراغ زبان همگذاری بروم و در واقع یاد گرفتم که رایانه چیست.

سیپیل: آیا آن‌ها در آن آزمایشگاه به شما زبان همگذاری را آموزش داده بودند؟

تامپسون: نه.

سیپیل: تجربه بعدی شما در زمینه برنامه‌نویسی چه بود؟

تامپسون: آن رایانه G15 مفسری به نام اینترکام ۵۰۱ داشت. ما تمرینات دروس مهندسی برق را باید به زمان اینترکام برنامه‌نویسی می‌کردیم. یکی از دانشجویان تحصیلات تکمیلی که دوست من بود، مفسری برای اینترکام بر روی یک ماشین بزرگ آی‌بی‌ام که در سرتاسر پردیس دانشگاه از آن استفاده می‌کردند، نوشت. من لیست برنامه‌اش را گرفتم و در تعطیلات کریسمس آن را خواندم و تجزیه و تحلیل کردم. من زبانی که برنامه به آن نوشته شده بود را نمی‌دانستم. در واقع، زبان نلیاک^{۱۹} بود. برنامه دوستم بسیار عالی نوشته شده بود و من بدین ترتیب برنامه نویسی را یاد گرفتم. هم زبان نلیاک را، هم اینترکام را و هم چگونگی تفسیر را. تقریباً یک هفته تمام فقط نشسته بودم و کد برنامه را می‌خواندم. بعد برگشتم و پرسش‌هایی در مورد آن برنامه از دوستم پرسیدم. چند خطای کوچک هم در آن پیدا کرده بودم. پس از آن، دیگر یاد گرفته بودم چگونه برنامه بنویسم و خیلی هم در این زمینه مهارت پیدا کردم. و بعد کارهای برنامه‌نویسی گرفتم. من در طول دوران تحصیلم به شیوه خاص خودم کار می‌کردم: کار-مطالعه-و شغل‌های عجیب. من دستیار پژوهشی یکی از دانشجویان تحصیلات تکمیلی بودم و کارهای برنامه‌نویسی مربوط به رساله‌اش را انجام می‌دادم. همچنین دستیار آموزشی یکی از استادان بودم. و همزمان برای مرکز رایانه دانشگاه برنامه‌نویسی می‌کردم. بخشی از کارم در مرکز رایانه این بود که در یک اتاق می‌نشستم و برنامه‌های دانشجویان را اشکال زدایی می‌کردم. آن‌ها پیش من می‌آمدند و می‌گفتند «فقط یک چیز را تغییر دادم» و من می‌گفتم «خوب، بگذار نگاهی به آن یک چیز بیندازیم و ببینیم چه اتفاقی افتاده است».

سیپیل: آیا این کارها باعث پرورش مهارت‌های اشکال زدایی شما شد یا فقط یک کار احمقانه و ناخوشایند بود؟

تامپسون: نوعی از اشکال زدایی را برایم آسان کرد. پس از آن، خطاهای متداول را خیلی خوب و سریع درک می‌کردم. خیلی از

دانشجویان چند روز بر روی برنامه‌هایشان کار کرده بودند و وقتی پیش من می‌آمدند من فوراً دست روی نقطه خطایشان می‌گذاشتم! **سیپیل:** مدرک شما در مهندسی برق بود؟ آیا در آن زمان در دانشگاه شما مدرک دانش رایانه (علوم کامپیوتر) هم می‌دادند؟

تامپسون: نه. در آن زمان در سرتاسر آمریکا دانش رایانه تازه داشت سر برمی‌آورد. و این به دو شیوه بود. یا به صورت نظری در دانشکده‌های ریاضی و یا به صورت عملی در دانشکده‌های مهندسی برق. در برکلی در آن زمان دانش رایانه تقریباً منحصرأ درون رشته مهندسی برق بود. دانشکده ریاضی هم سعی می‌کرد اما از لحاظ سیاسی به قدر کافی ز رنگ و موقع‌شناس نبودند که بتوانند با دانشکده مهندسی برق رقابت کنند. **سیپیل:** دانشگاه برکلی سرانجام به خاطر چیزهایی مثل آزمایشگاه سیستم‌ها معروف شد- یعنی ساختن چیزها- و نه به خاطر مشارکت در ایجاد و توسعه نظریه‌ها.

تامپسون: بله، دقیقاً همین‌طور است. و این آغاز پیدایش دانشکده‌های دانش رایانه نظری، نظیر دانشگاه کورنل، و دانشکده‌های دانش رایانه به سبک برکلی بود. من یکسالی را هم در دوره تحصیلات تکمیلی در آنجا گذراندم. نه به دلیل آن که هدف خاصی داشتم بلکه به دلیل آن که کار دیگری برای انجام دادن نداشتم و در آنجا هم اوقات خوشی داشتم.

سیپیل: بلافاصله بعد از دوره کارشناسی؟

تامپسون: بله. راستش را بخواهید من در دانشگاه کار می‌کردم و حتی برای دوره تحصیلات تکمیلی درخواست هم نداده بودم. یکی از استادان در واقع به جای من درخواست داده بود و بعد به من گفت که پذیرفته شده‌ام.

سیپیل: همچنان در رشته مهندسی برق؟

تامپسون: بله. سال آخر دوره کارشناسی و آن یکسال دوره تحصیلات تکمیلی برای من خیلی سرگرم کننده بود. هیچ کاری که دوست نداشتم را انجام ندادم. هیچ پیش‌نیازی وجود نداشت، هیچ چیز. من برای فارغ‌التحصیل شدن در ترم تابستانی درس تاریخ آمریکا یا یک چیزی مثل آن را گرفتم که برای گرفتن مدرک لازم بود. اما بجز آن، در سال آخر و در دوره کارشناسی ارشد، من نصف درس‌هایی که گرفته بودم را درس می‌دادم.

نظریه‌های اصلی راینش کم‌کم داشت شکل می‌گرفت. مرتب کردن شل^{۲۰} تازه ارائه شده بود و هیچکس نمی‌توانست درک کند که چرا این سریعتر از روش‌های مرتب کردن از رتبه آماریⁿ است. و همه بر روی آن آزمایش می‌کردند و سعی می‌کردند حکمتش را بیابند. به راحتی می‌شد دید که مرتب می‌کند اما هیچکس نمی‌دانست که چرا سریع است و چرا از رتبهⁿ به توان $1/3$ یا چیزی شبیه آن است. و این یک عدد طبیعی نبود. و از این‌جا تمام این صحبت‌های مربوط به رتبه آماری الگوریتم‌ها بیرون آمد. و همین‌طور نخستین الگوریتم‌های مرتب کردن از رتبه^{n log n}. دوره هیجان‌انگیز و جالبی بود.

17- setup

18- interpreted language

19- NELIAC

20- shell sort



تامپسون: بله، درست است. من آن را بر روی آی بی ام ۱۶۲۰ که در دانشکده فیزیک قرار داشت اجرا کردم. من همه رایانه‌های دانشگاه را می‌دانستم کجا هستند و شب‌ها برنامه‌هایم را روی آن‌ها اجرا می‌کردم. به علاوه، در مرکز رایانه اصلی دانشگاه هم ۲۰ شماره حساب مختلف داشتم. در نتیجه، هیچگاه محدودیت سخت‌افزاری پیدا نمی‌کردم.

سیپل: برنامه شما براساس جستجوی همه حالت‌ها بود؟

تامپسون: بله.

سیپل: این برنامه هم به زبان همگذاری (اسمبلی) بود؟

تامپسون: باید فکر کنم. بله، احتمالاً به زبان همگذاری بود. دقیقاً به یاد نمی‌آورم.

سیپل: شما باید زبان فرتن را هم یک زمانی یاد گرفته باشید.

تامپسون: بله، من باید در مرکز رایانه، فرتن درس می‌دادم و برنامه‌های فرتن را اشکال زدایی می‌کردم. من هیچگاه به این زبان برنامه‌نویسی نکرده‌ام. آن اوایل، یک مترجم^{۲۱} فرتن برای یونیکس نوشتیم. زبان B هم در همین راستا به دست آمد.

سیپل: من فکر می‌کردم زبان B ترجمه شما از BCPL است.

تامپسون: به نوعی بود. هنگامی که آن را شروع کردم، نمی‌دانستم چه بود. از نظر معنایی^{۲۲} شبیه BCPL شد. هنگامی که شروع کردم قرار بود شبیه فرتن باشد. در آن مقطع من نخستین توصیف BCPL را نوشتم. و از قواعد معنایی واضح و روشن آن خوشم آمد. و به همین خاطر بود که فرتن را کنار گذاشتم و از نظر قواعد نحوی^{۲۳} اساساً همان C و از نظر معنایی BCPL شد.

سیپل: آیا از زمانی که برنامه‌نویسی یاد گرفته‌اید تا کنون، اختلاف فاحشی در طرز تفکر شما درباره برنامه‌نویسی به وجود آمده‌است؟ آیا فکر می‌کنید شیوه برنامه‌نویسی شما بلوغ بیشتری یافته‌است؟ آیا چیزهایی یاد گرفته‌اید که شما را به گذشته برگرداند و بگویید «اوه، آن موقع نمی‌دانستم چه کار می‌کردم»؟

تامپسون: نه، واقعاً نه. گاهی اوقات به کارهایی که قبلاً انجام داده بودم نگاه می‌کنم و می‌گویم «آن وقت‌ها خیلی بهتر بودم.» من از آن یک هفته‌ای که برنامه دوستم را خواندم و با مفهوم برنامه‌نویسی آشنا

من در برکلی دوستان زیادی داشتم. تعدادی استاد جوان، به ویژه یک استاد ریاضی که رابطه نزدیکی با او داشتم و یک استاد مهندسی برق، و نیز آن دانشجوی دکتری که برایش کار می‌کردم و چند نفر دیگر. آن‌ها برای من کلاسی را ابداع می‌کردند و من آن را درس می‌دادم. **سیپل:** آیا شما به طور رسمی به عنوان دانشجو در آن کلاس ثبت نام می‌کردید یا آن که در واقع به عنوان استاد درس نامتان ذکر می‌شد؟ **تامپسون:** نه، نه. به عنوان استاد نامم در جایی درج نمی‌شد. همه آن درس‌ها در قالب درس EE199 بود. یعنی پژوهش انفرادی یا گروهی. آن‌ها کلاسی را ابداع می‌کردند و عنوانی برایش می‌گذاشتند و تدریس آن را بر عهده من می‌گذاشتند. معمولاً سه، چهار دانشجو هم در این درس‌ها ثبت نام می‌کردند.

سیپل: و از نظر دانشگاه و به طور رسمی یکی از آن دانشجویان شما بودید؟

تامپسون: بله.

سیپل: آیا تدریس را دوست داشتید؟

تامپسون: تا حدی. من دوبار برگشته‌ام و درس داده‌ام. یکسال در برکلی، سال‌های ۷۶-۱۹۷۵، و یکسال در سیدنی، سال ۱۹۸۸. برای من خیلی جالب بودند و برای خودم هم جنبه آموزشی داشتند زیرا من هیچگاه تحصیلات دانشگاهی در رشته دانش‌رایانه نداشتم. استادان میهمان معمولاً یک درس می‌دادند. من پنج کلاس را درس می‌دادم. بعضی از کلاس‌ها را دو بار درس دادم و فکر می‌کنم که آن‌ها بهترین تجربه برایم بودند زیرا سال اول خودم یاد گرفتم و بار دوم می‌دانستم که دارم چه می‌گویم و می‌توانستم مطالب درسی را به نحو سازمان یافته‌تری ارائه کنم و دو گام از دانشجویانم جلوتر باشم. اما تدریس برای بار سوم دیگر خیلی کسالت‌بار بود. من یکبار این کار را کردم که کاملاً اشتباه بود. بنابراین، من هیچگاه نمی‌توانستم معلم شوم چون شما باید یک کلاس را بارها و بارها درس بدهید. من هرگز نمی‌توانم چنین کاری بکنم. اما من عاشق تدریسم: کار سخت برای اولین کلاس و لذت دومین کلاس. سومی به بعد فاجعه است.

سیپل: نخستین برنامه جالبی که نوشتید چه بود؟

تامپسون: نخستین برنامه محاسباتی طولانی که نوشتم، حل مسئله پنتامینو بود (پنج مربع را در نظر بگیرید. آن‌ها را می‌توان به طرق مختلفی از یک ضلع به هم چسباند. بدین ترتیب شکل‌های مختلفی پدید می‌آید. ۱۲ تا از این شکل‌ها کم و بیش شبیه حروف انگلیسی هستند. مثلاً اگر هر پنج مربع را زیر هم قرار دهیم شکل حرف I می‌شود و اگر سه تا را به طور ردیفی کنار هم قرار دهیم و دوتای دیگر را از مربع وسطی به سمت پایین بگذاریم، شکل حرف T می‌شود. مسئله پنتامینو یا پنج تکه‌ای‌ها این است که یک جدول بزرگ را باید با کنار هم قرار دادن این ۱۲ شکل بپوشانیم به نحوی که هیچ خانه جدول خالی نماند. مترجم). با این مسئله آشنا هستید؟

سیپل: پر کردن خانه‌های جدول با پنج تکه‌ای‌ها؟ شبیه بازی تتریس؟

21- compiler
22- semantically
23- systax

شدم تا ۳۰، ۳۵ سالگی، خط به خط تمام برنامه‌هایی که نوشته بودم را با جزئیات کامل می‌دانستم. من در طول روز برنامه می‌نوشتیم و شب‌ها می‌نشستم و آن را خط به خط می‌خواندم و خطاهایش را پیدا می‌کردم. روز بعد برمی‌گشتم و مطمئناً هنوز خطاهایی باقی‌مانده بود. سیبل: فکر می‌کنید که وقتی ۳۵ ساله بودید هنوز برنامه‌هایی که ده سال پیشتر نوشته بودید را به یاد می‌آوردید؟

تامپسون: بله، کاملاً. بعد از آن، نه همه چیزها بلکه برخی از آن‌ها را به یاد می‌آوردم.

سیبل: آیا در مورد یادگیری برنامه‌نویسی، چیزی هست که دلتان می‌خواست جور دیگری انجام داده بودید؟ آیا هیچ پشیمانی از مسیری که پیموده‌اید دارید یا چیزی هست که آرزو می‌کنید زودتر انجام داده بودید؟

تامپسون: اوه، مطمئناً، مطمئناً. ای کاش در دبیرستان، درس تایپ کردن را برداشته بودم. من تا همین حالا هم از ضعفی که در این مورد دارم در عذابم. من برای کارهایم هیچ نوع برنامه‌ریزی نداشتیم. نظم و انضباط نداشتیم. همیشه هر کاری که دلم می‌خواست را انجام می‌دادم. اگر آینده‌نگری یا برنامه‌ریزی داشتیم، چیزهایی مثل تایپ کردن بود که وقتی فرصتش پیش می‌آمد انجام می‌دادم. دیگر این که درس‌های ریاضی بیشتر و عمیق‌تری برمی‌داشتیم زیرا چند بار به دلیل ضعف دانش ریاضی، بعضی کارها را کنار گذاشته‌ام. بنابراین، در پاسخ به سؤال شما باید بگویم بله، چیزهای کوچکی وجود داشته است. اما اگر به گذشته برگردم و مجبور باشم همین کارها را دوباره انجام دهم، با شناختی که از خودم دادم مطمئنم که هیچ کاری را به نحو دیگری انجام نخواهم داد. من اساساً برای هیچ کاری برنامه‌ریزی نمی‌کنم و فقط گام بعدی را برمی‌دارم.

سیبل: شما پس از فارغ‌التحصیلی مستقیماً به استخدام آزمایشگاه‌های بل درآمدید. این امر چگونه اتفاق افتاد؟ به نظر نمی‌رسد که شما در آن مقطع یک پژوهشگر کلاسیک دانشگاهی بودید.

تامپسون: من به آنجا سوق داده شدم. توضیحش کمی مشکل است. یکی از استادانم که دوستی نزدیکی هم با من داشت، نظر مسئول استخدام آزمایشگاه‌های بل را به من جلب کرده بود. اما من دنبال کار نبودم و مطلقاً هیچ نوع بلندپروازی و جاه‌طلبی نداشتیم. استادم برای من وقت ملاقات گرفت و من در نخستین ملاقات به مسئول استخدام آزمایشگاه‌های بل گفتم که علاقه‌ای به استخدام شدن در آنجا ندارم. اما او مرا رها نکرد. مدتی بعد به من تلفن کرد و گفت که می‌خواهد پیش من بیاید و مرا ببیند. او به خانه ما آمد و گفت که می‌خواهد من برای مصاحبه به آزمایشگاه‌های بل بروم. من به او جواب منفی دادم. او گفت «این یک مسافرت مجانی است. تو می‌توانی آنجا هر کاری دلت خواست بکنی.» و من گفتم «همان‌طور که پیش از این گفتم به کار در آنجا علاقه‌ای ندارم اما خوشحال می‌شوم که یک مسافرت مجانی به ساحل شرقی داشته باشم زیرا دوستان زیادی آنجا دارم و می‌توانم به ملاقاتشان بروم.» بدین ترتیب من به آزمایشگاه‌های بل

رفتم و دو روز را در آنجا گذراندم و سپس ماشینی کرایه کردم و در ساحل شرقی بالا و پایین رفتم و با دوستان دبیرستانی‌ام ملاقات کردم. **سیبل:** حتماً افراد آزمایشگاه‌های بل در شما چیزی دیده بودند که اصرار داشتند شما را به آنجا ببرند.

تامپسون: من نظر آن‌ها را نمی‌دانم اما از نقطه‌نظر خودم آن‌ها کسانی بودند که من در کلاس‌هایی که درس می‌خواندم یا درس می‌دادم، مقالاتشان را خوانده بودم. و آن‌ها را به اسم و شهرتشان می‌شناختم. و هنوز هم کارهای جالبی می‌کردند. برای من کار، کار بود و به نظر من این افراد کار جدی نمی‌کردند. بلکه اوقات خوشی در آنجا داشتند. درست مثل مدرسه.

سیبل: در بدو ورود به آنجا چه نوع کاری می‌کردید؟ **تامپسون:** آزمایشگاه‌های بل درگیر پروژه مالتیکس بود و من نیز برای کار در همین پروژه استخدام شدم. اما پس از مدتی آزمایشگاه‌های بل به این نتیجه رسید که مالتیکس مناسب آن‌ها نیست و از پروژه کنار کشید.

اما ماشین‌های مالتیکس هنوز در آنجا بودند. و من و یکی دو نفر دیگر برای حدوداً یکسال با آن‌ها کار می‌کردیم. بدین ترتیب بود که من کار در زمینه سیستم‌عامل را آغاز کردم و شروع به نوشتن بخش‌های کوچکی از سیستم‌عامل نمودم. سیستم‌عامل مالتیکس بسیار پیچیده بود و کار با آن عذاب‌آور. اما من تا وقتی که این ماشین‌ها آنجا بودند با آن‌ها کار می‌کردم. سرانجام این ماشین‌ها بیرون برده شدند و کار من با آن‌ها به پایان رسید. خوشبختانه من توانستم چند ماشین بدون استفاده دیگر در آنجا پیدا کنم و سیستم‌عامل یونیکس را در اصل بر روی همین ماشین‌های بسیار کند پی دی پی نوشتم.

سیبل: آیا رؤسای شما می‌دانستند که سرگرم چه کاری هستید و برای انجام این پروژه پژوهشی وقت کافی در اختیارتان می‌گذاشتند یا این که صرفاً به دلیل آن که هنوز کار جدیدی به شما داده نشده بود وقت انجام آن را داشتید؟

تامپسون: نه. راستش را بخواهید من اصلاح شدنی نبودم. فکر می‌کردم که نهایتاً اخراجم خواهند کرد اما برایم مهم نبود. قرار بود که ما پژوهش‌های بنیادی انجام دهیم اما برخی پژوهش‌های بنیادی وجود داشت که ما باید انجام می‌دادیم و برخی پژوهش‌های بنیادی وجود داشت که ما نباید انجام می‌دادیم. و از تجربه نافرجام و پرهزینه مالتیکس چنین نتیجه‌گیری شده بود که سیستم‌های عامل جزء آن دسته از پژوهش‌های بنیادی هستند که ما نباید انجام می‌دادیم. به همین خاطر، انتظار داشتیم که به خاطر کاری که در حال انجام دادنش بودم، نهایتاً اخراج شوم. اما چنین نشد.

سیبل: شما چگونه طراحی نرم‌افزار را انجام می‌دهید؟ آیا ابتدا به رسم نمودارها و استفاده از ابزارهای UML می‌پردازید یا آن که مستقیماً به سراغ کدنویسی می‌روید؟

تامپسون: بستگی به اندازه و بزرگی نرم‌افزار دارد. در اغلب موارد، طراحی برای یک دوره زمانی درون ذهن من انجام می‌گیرد- نه روی کاغذ- و

آن که کدنویسی را آغاز کردید، چقدر به این ساختار پایبند می‌مانید؟ **تامپسون:** من به کد نمی‌چسبم. اگر پس از آن که نصف برنامه‌نویسی را انجام داده‌ام، روش بهتری برای پیاده‌سازی بیابم، همه آن را کنار می‌گذارم و دوباره از نو شروع می‌کنم. افراد زیادی را می‌شناسم که وقتی یک خط کد نوشتند تا آخر عمرشان به آن دست نمی‌زنند، مگر آن که خطا داشته باشد. به ویژه اگر روالی²⁴ را با یک API (رابط برنامه کاربردی) نوشته باشند و آن API را جایی مثلاً پشت یک پاکت طراحی کرده باشند، دیگر همین است که هست. هرگز آن را تغییر نمی‌دهند، هر چقدر هم که بد باشد. من برعکس، همیشه مشتاقم که راه بهتری بیابم و کارهای قبلی را پاره کنم و دور بریزم. من هیچگاه عاشق کدهایی که نوشته‌ام نبوده‌ام. کد به خودی خود فاسد می‌شود و باید دوباره نویسی شود. حتی اگر هیچ چیز تغییر نکرده باشد، به دلایلی کد فاسد می‌شود. **سیبل:** چگونه تصمیم می‌گیرید که کد باید دور انداخته شود؟ **تامپسون:** وقتی که کار کردن با آن سخت باشد. من این کار را سریعتر از اغلب افراد انجام می‌دهم. به محض آن که بخواهم چیزی به کد اضافه کنم و احساس کنم که کاری که باید انجام دهم بسیار مشکل است، آن کد را دور می‌اندازم و دوباره از نو شروع می‌کنم و به طراحی جدیدی می‌رسم که کاری که می‌خواستم انجام دهم، به سادگی انجام شود. **سیبل:** آیا این قضیه در مورد کار کردن با کد دیگران نیز صادق است؟

تامپسون: بستگی به این دارد که اختیارش دست من باشد یا نه. اگر اختیار داشته باشم مطمئناً همین کار را می‌کنم. اما اگر چنین اختیاری نداشته باشم. یا باید تحمل کنم و یا اصلاً قیدش را بزنم و آن کار را انجام ندهم.

سیبل: در حالتی که بخواهید کد فرد دیگری را کنار بگذارید و دوباره خودتان آن را بازنویسی کنید این خطر وجود دارد که نکات ظریفی از چشمان دور بماند و یا از برخی جزئیات کار کردی آن غفلت کنید. آیا تاکنون چنین مشکلی برایتان پیش آمده است؟

تامپسون: خوب، البته ممکن است این اتفاق بیفتد اما این هم بخشی از فرایند اشکال‌زدایی است. اگر چیزی از چشمتان دور مانده باشد، هر وقت به آن پی بردید، انجامش می‌دهید. همان‌طور که گفتم این بخشی از فرایند اشکال‌زدایی است. همیشه اولین نسخه، کامل نیست و نیاز به توسعه دارد.

سیبل: هنگامی که نرم‌افزاری را نوشتید آیا برمی‌گردید و آن را مستند می‌کنید؟

تامپسون: بستگی به این دارد که آن نرم‌افزار برای چه منظوری نوشته شده است. اگر برای خودم باشد، نه این کار را نمی‌کنم. یک جمله توضیحی²⁵ خیلی خیلی کوتاه در ابتدای هر بخش می‌گذارم که کار آن بخش را به طور کلی توضیح دهد. اما اگر بخشی از یک سیستم بزرگتر باشد یا چیزی باشد که قرار است منتشر شود، در آن

من بر روی قسمت‌های سخت آن تمرکز می‌کنم. بخش‌های ساده نیازی به طراحی ندارند و هر وقت آماده باشم مستقیماً آن‌ها را می‌نویسم. اما به بخش‌های سخت، مدتی فرصت می‌دهم تا درون ذهنم جوانه بزنند و برویند. بعد لحظه‌ای می‌رسد که تکه‌ها شروع به کامل شدن و بیرون ریختن می‌کنند و می‌توانم ببینم که هرم از پایین به بالا از این تکه‌ها در حال ساخته شدن و تکمیل است. هنگامی که ارتفاع هرم در ذهنم به قدر کافی بلند شد، آنگاه کدنویسی را از لایه‌های پائینی آغاز می‌کنم. **سیبل:** اما شما فقط برگ‌ها را نمی‌سازید بلکه ساختاری که قرار است آن‌ها در آن جای گیرند را نیز در ذهن دارید.

تامپسون: من با دیدن تکه‌های پائینی هرم می‌توانم ساختار برنامه را در ذهنم مجسم کنم. مشکل، تجزیه آن و به دست آوردن تکه‌های پائینی است. یکی از مشکلاتی که من با برنامه‌نویسی مدرن دارم نیز همین است. لایه پس از لایه پس از لایه ساخته می‌شود و این لایه‌ها در واقع هیچ کاری بجز «ترجمه» انجام نمی‌دهند. همیشه خواندن بالا به پایین²⁶ یک برنامه مرا گیج می‌کند. از بالا که برنامه را می‌خوانیم می‌گویید «کاری را انجام بده.» و شما دنبال آن «کار» می‌گردید و وقتی پیدایش کردید و آن را خواندید می‌بینید که می‌گوید «یک کار دیگر را انجام بده» و این روند همین‌طور دنبال می‌شود و شاید دوباره به ابتدای برنامه برگشتید. صرفاً تنزل دادن مسأله به سطحی عمیقتر و عمیقتر است. من نمی‌توانم آن را به خاطر بسپارم. در کش نمی‌کنم. **سیبل:** چرا در این حالت هم پایین به بالا²⁷ نمی‌خوانید؟ برگ‌ها هم یکجایی همان جاها هستند.

تامپسون: خوب، شما نمی‌دانید چه چیزهایی برگ‌ها هستند و چه چیزهایی نیستند. اگر مستندات خوبی داشته باشد شما می‌توانید آن را بخوانید و دیگر نیازی به خواندن کد نیست. اما اگر فقط کد را در اختیارتان قرار دهند و به شما بگویند «آن را بخوان و سعی کن بهترش کنی یا سعی کن کار دیگری انجام دهد،» آنگاه معمولاً چاره‌ای جز خواندن بالا به پایین نیست. من که همین کار را می‌کنم. **سیبل:** آیا شما هیچگاه پیش از شروع به نوشتن کد، چیز دیگری را روی کاغذ نوشته‌اید؟

تامپسون: من معمولاً پیش از نوشتن کد، ساختمان داده‌ها²⁸ را می‌نویسم. معمولاً الگوریتم، روندنما²⁹ یا چیزهایی از این قبیل را روی کاغذ نمی‌آورم. بجز چیزهایی که تقریباً در هر خط برنامه باید به آن‌ها رجوع کرد، یعنی ساختمان داده‌ها.

سیبل: منظورتان این است که مثلاً وقتی می‌خواهید برنامه C بنویسید، کد C مربوط به تعریف ساختمان داده‌ها را می‌نویسید؟

تامپسون: نه. جعبه‌های کوچک با پیکان‌ها و چیزهایی از این قبیل. **سیبل:** شما گفتید که پیش از آغاز کردن کدنویسی، ساختار کلی برنامه، یا به قول خودتان هرم، را در ذهنتان به دست می‌آورید. پس از

24- top-down

25- bottom-up

26- data structures

27- flowchart

28- routine

29- comment

صورت برای مستندسازی آن وقت می‌گذارم. در غیر این صورت، نه. مستندسازی، هنری است به زیبایی برنامه‌نویسی. من به ندرت مستنداتی را دیده‌ام که از آن خوشم آمده باشد. معمولاً جزئیات خیلی بیشتری از آنچه باید داشته باشد دارد و شامل مقدار زیادی مطالب نامربوط است. مستندسازی کار بسیار سخت و زمان‌گیری است و اگر بخواهید آن را درست انجام دهید باید با آن مثل برنامه‌نویسی رفتار کنید. باید آن را به بخش‌های مختلف تجزیه کنید، بخش‌ها را به نحو قشنگی کنار هم قرار دهید، و هرگاه اشتباه بود آن را دوباره‌نویسی کنید. مردم معمولاً این کار را نمی‌کنند.

من همچنین مستندسازی پایین به بالا را ترجیح می‌دهم و معمولاً این روشی نیست که برای نوشتن مستندات به کار می‌رود.

سیبل: پس شما دوست دارید که با همان ترتیبی درک کنید که دوست دارید آن را بنویسید، یعنی پایین به بالا؟

تامپسون: بله، این روشی است که می‌توانم آن را به خاطر بسپارم. در غیر این صورت، ممکن است کد را بخوانم و بلافاصله آن را درک کنم اما از خاطرم می‌رود. اگر ساختارش را درک کنم، آنگاه جزئی از وجودم می‌شود و آن را خواهم فهمید.

سیبل: شما در سخنرانی خود هنگام دریافت جایزه تورینگ خاطرنشان کردید که اگر دن بابرو^{۳۰} مجبور می‌شد به جای استفاده از رایانه قدرتمند پی‌دی‌پی ۱۰ از رایانه پی‌دی‌پی ۱۱ استفاده کند، احتمالاً او به جای شما و دنیس ریچی جایزه را دریافت می‌کرد.

تامپسون: من فقط می‌خواستم بگویم که ما خوش‌شانس بودیم.

سیبل: آیا فکر می‌کنید که استفاده از یک رایانه کم‌قدرت‌تر به نفعتان بوده است؟

تامپسون: این که پی‌دی‌پی ۱۱ کوچک و کارآمد بود مطمئناً به نفع ما بود. البته من فکر می‌کنم که ما آن کد را به هر حال می‌نوشتیم. خوش‌شانسی ما این بود که درست در اوج انقلاب رایانه‌های کوچک^{۳۱} قرار داشتیم. پی‌دی‌پی ۱۰ یک رایانه بزرگ بود که تحت سرپرستی مرکز رایانه کار می‌کرد. این که رایانش از حالت متمرکز درآمد به نظر من واقعاً یک خوش‌شانسی بود.

سیبل: آیا یونیکس نیز از این که به زبان C نوشته شده بود سود نبرد، در حالی که سیستم‌عامل‌هایی چون TENEX و ITS به زبان همگذاری (اسمبلی) نوشته شده بودند و نتوانستند به راحتی یونیکس در سخت‌افزارهای دیگر به کار گرفته شوند؟

تامپسون: زبان‌های برنامه‌نویسی سیستم^{۳۲} خوبی هم وجود داشت که تا حدی مورد استفاده قرار گرفته بودند.

سیبل: مثلاً؟

تامپسون: نلیاک نسخه برنامه‌نویسی سیستم الگول ۶۸ بود.

سیبل: بلیس^{۳۳} هم مال همان دوره بود؟

تامپسون: فکر می‌کنم بعد از آن بود. و تأکیدش بر ترجمه عالی بود. من فکر می‌کنم از ابتدا روشن بود که نباید خودتان را برای آن بکشید. می‌توانید خوب عمل کنید اما نه عالی. و دلیلش این است که زمانی که طول می‌کشد تا از خوب به عالی برسید، قانون مور^{۳۴} شما را پشت سر گذاشته است. شما می‌توانید کارکرد را ۱۰ درصد بهبود بخشید اما تا وقتی که این ۱۰ درصد بهبود را به دست آورید، رایانه‌ها دو برابر سریع‌تر شده‌اند. من فکر می‌کنم تلاشی که برای رسیدن به حد عالی صورت می‌گیرد بیهوده و اتلاف وقت است. کار بسیار سختی است. شما معمولاً به همان تعداد خطایی که برطرف می‌کنید، خطای تازه به وجود می‌آورید. باید دست بردارید و آن ۱۰۰ درصد زمان اضافی را برای انجام ۱۰ درصد کار نگیرید.

سیبل: شما احتمالاً مقاله ریچارد گابریل با عنوان «بدتر، بهتر است» را خوانده‌اید.

تامپسون: نه.

سیبل: او آنچه سبک ام‌آی تی نامیده است، که در آن صحت و درستی، تمام چیزهای دیگر را تحت‌الشعاع قرار می‌دهد، و سبک نیوجرسی (یعنی آزمایشگاه‌های بل)، که در آن سادگی پیاده‌سازی بالاترین ارزش را دارد، را با هم مقایسه کرده است. نظریه او این بود که سبک نیوجرسی، که او آن را «بدتر، بهتر است» نامیده، این امکان را به وجود می‌آورد که زودتر چیزی آماده و اجرایی شود و از آن نقطه به بعد، بهبود خواهد یافت.

تامپسون: من فکر می‌کنم ام‌آی تی همیشه نسبت به یونیکس عقده حقارت داشته است. من یک سخنرانی در ام‌آی تی درباره یونیکس داشتم و فکر می‌کنم مایکل درتوزوس مرا معرفی کرد. او توضیح داد که چرا یونیکس در ام‌آی تی نوشته نشده و چرا باید این‌گونه می‌شده است. چرا با وجودی که آن‌ها فرصت داشتند، افراد مناسب داشتند، همه چیز داشتند، این کار در آنجا انجام نشد؟ به نظر من علتش این بود که آن‌ها رقبایی در ذهن داشتند. در ذهن من در آن موقع چنین چیزی نبود. ما یونیکس را درست کردیم و آن‌ها مالتیکس را، که خیلی بزرگ و عظیم بود. به نظر من این به وضوح از علائم سیستم دوم است. **سیبل:** منظورتان این است که مالتیکس پس از CTSS (سیستم اشتراک زمانی سازگار^{۳۵})، سیستم دوم بود؟

تامپسون: بله، همه چیزش بیش از حد بزرگ بود. آنقدر بزرگ که تقریباً غیرقابل استفاده. آن‌ها هنوز هم ادعا می‌کنند که موفقیت بزرگی بود اما واضح است که چنین نبود.

سیبل: درک من این است که افراد ام‌آی تی هم چنین نظری نسبت به مالتیکس داشتند. آن‌ها ITS و سیستم‌های بر پایه لیسپ که ساختند را ترجیح می‌دادند. به نظر می‌رسد که راه آن‌ها پس از مالتیکس شاخه شاخه شد. یونیکس بیرون آمده بود، همان‌طور که شما به خوبی می‌دانید، و از طرف دیگر در ام‌آی تی، دستداران لیسپ

34- Moore

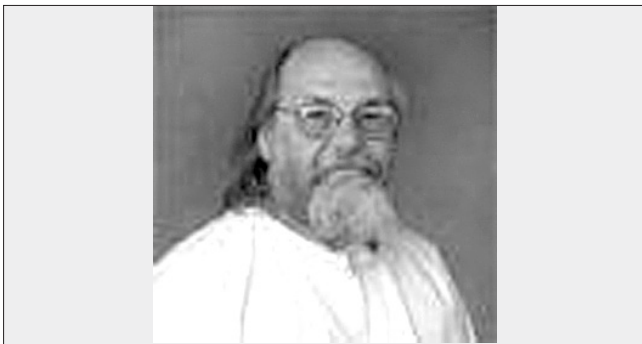
35- Compatible Time-Sharing System

30- Dan Babrow

31- minicomputers

32- system programming

33- Bliss



نوشتن برنامه‌هایی برای به کار انداختن سیستم پرونده بود. من یک چیز تعاملی^{۴۰} می‌خواستم.

سیبیل: پس شما فقط می‌خواستید یک سیستم پرونده بنویسید؟ در آن موقع برای نوشتن یک سیستم عامل برنامه‌ریزی نکرده بودید؟ **تامپسون:** همین‌طور است. در ابتدا فقط یک سیستم پرونده بود. اما در میانه‌های راه متوجه شدم که یک سیستم اشتراک زمانی واقعی است. من داشتم پوسته‌ای برای راه‌اندازی سیستم پرونده می‌نوشتم و پس از آن داشتم چند برنامه دیگر برای این کار می‌نوشتم و در همین موقع بود که به خودم گفتم «تنها چیز دیگری که نیاز دارم تا یک سیستم عامل کامل داشته باشم، ویراستار^{۴۱} است.»

سیبیل: بدترین خطاهایی که تاکنون رفع کرده‌اید چه بوده است؟ **تامپسون:** اساساً خطاهایی که مربوط به خراب شدن محتویات حافظه هستند. این خطاها دیگر اتفاق نمی‌افتند و من علتش را نمی‌دانم. اما در آن روزهای اولیه ما همیشه با سخت‌افزارهای آزمایشی کار می‌کردیم و این نوع خطاهای سخت‌افزاری وجود داشت.

سیبیل: منظورتان این است که محتویات حافظه فقط به علل سخت‌افزاری خراب می‌شد و نه مثلاً به دلیل یک اشاره‌گر^{۴۲} از کنترل خارج شده؟

تامپسون: می‌توانست اشاره‌گر باشد، می‌توانست سخت‌افزار باشد، یا ترکیبی از آن‌ها. خطایی که اکنون دارم درباره‌اش فکر می‌کنم و احتمالاً بدترین خطا بود بر روی پی‌دی‌پی ۱۱ اتفاق افتاد. این ماشین، عمل ضرب نداشت و شما می‌توانستید یک واحد مستقل برای ضرب بخرید و آن را به عنوان یک دستگاه جانبی ورودی-خروجی به سیستم وصل کنید. شما صورت و مخرج کسر را ذخیره می‌کردید و دستور تقسیم را می‌دادید. سپس جواب به صورت خارج قسمت و باقی‌مانده در می‌آمد. این واحد برای پی‌دی‌پی ۱۱ بدون مدیریت حافظه ساخته شده بود و ما نخستین سخت‌افزار آزمایشی پی‌دی‌پی ۱۱ با مدیریت حافظه را گرفته بودیم و این واحد ضرب به خوبی با مدیریت حافظه جفت و جور نمی‌شد.

من برنامه‌ای نوشته بودم برای ثبت رکورد جهانی جدیدی برای تعداد ارقام e. رکورد جهانی قبلی محدودیت محاسباتی نداشت بلکه محدودیت ورودی-خروجی داشت. من به الگوریتم تازه‌ای دست

کارهایشان را روی پی‌دی‌پی ۱۰ انجام می‌دادند و سیستم‌های بر پایه‌ی لیسپ را ساختند که نهایتاً به ماشین‌های لیسپ منجر شد.

تامپسون: بله، بله. من همه آن‌ها را می‌شناختم. به نظر من یک کار احمقانه بود. من فکر نمی‌کردم زبان لیسپ آنچنان زبان منحصر به فرد و جامعی باشد که نیاز به ساختن ماشین خاص داشته باشد. و فکر می‌کنم درستی نظرم اثبات شد. من همیشه می‌گفتم که این یک کار احمقانه است. پی‌دی‌پی ۱۱ ماشین خیلی خوبی برای اجرای لیسپ بود. پی‌دی‌پی ۱۰ هم همین‌طور. نیازی به ساختن یک ماشین لیسپ نبود که تازه آن هم سریعتر از آن ماشین‌ها نبود. هیچ دلیلی برای ساخت ماشین لیسپ وجود نداشت. همان‌طور که گفتم کار احمقانه‌ای بود.

سیبیل: آیا ویژگی‌هایی در مالتیکس وجود داشت که شما آن‌ها را دوست داشتید اما هیچگاه در یونیکس قرار داده نشدند؟

تامپسون: چیزی که من دوست داشتم، سیستم پایگانی^{۳۶} پرونده‌ها و پوسته^{۳۷} - یک فرایند جداگانه که می‌توانستید با فرایندی دیگر جایگزین سازید - بود. پیش از آن، تمام سیستم‌ها به نوعی دارای یک زمان پردازش توکار^{۳۸} بودند که به ازای هر فرایند اجرا می‌شد. اما در کار با پوسته، هر زمان شما چیزی تایپ می‌کردید، یک فرایند جدید به وجود می‌آورد و هر چیزی تایپ کرده بودید را اجرا می‌کرد و وقتی آن از بین می‌رفت، دوباره بازمی‌گشتید.

سیبیل: این‌ها چیزهایی هستند که شما در یونیکس هم قرار دادید. آیا چیز دیگری بود که اکنون پشیمان باشید از این که در یونیکس قرار ندادید.

تامپسون: نه.

سیبیل: تا آنجا که من درباره تاریخچه یونیکس خوانده‌ام، ظاهراً شما طراحی آن را از مدتی قبل انجام داده بودید. بعد همسر و فرزندان به مدت یکماه به مسافرت رفته بودند و شما با خود گفته بودید که «چه خوب، حالا کدش را می‌نویسم.»

تامپسون: بله. سه چهار نفر از ما دور هم نشسته بودیم و درباره یک سیستم پرونده صحبت کرده بودیم. تنها کسی که کمتر شناخته شده است رود کانادی نام داشت. در آن روزها در آزمایشگاه‌های بل امکانات رفاهی خیلی خوب بود. شما می‌توانستید یک شماره تلفن را بگیرید و از سرویس آوانویسی^{۳۹} استفاده کنید. پیام می‌گذاشتید و می‌گفتید که می‌خواهید پیامتان نوشته شود. روز بعد آن را بر روی کاغذ تحویل می‌گرفتید. و کانادی پس از آن که ما درباره سیستم پرونده صحبت کردیم و روی تخته‌سیاه چیزهایی نوشتیم، تلفن را برداشت، شماره گرفت و آنچه روی تخته‌سیاه نوشته شده بود را خواند.

روز بعد که کاغذها به دسکمان رسید تقریباً همان مستندات طراحی بود. بعد من آن سیستم پرونده را بر روی پی‌دی‌پی ۷ پیاده‌سازی کردم. بعد تصمیم گرفتم آن را آزمایش کنم. اما مشکلی که داشتم،

36- hierarchical

37- shell

38- built-in

39- transcription

40- interactive

41- editor

42- pointer

یافته بودم که محدودیت محاسباتی داشت اما ورودی و خروجی بدیهی و واضح بود. الگوریتم من به عملیات ضرب و تقسیم فوق‌العاده زیادی نیاز داشت و ما متوجه شدیم که هرگاه من برنامه‌ام را روی ماشین اجرا می‌کردم، محتویات حافظه‌اش به هم می‌ریخت. پیگیری‌ها و ردیابی خود ما نشان داد که اشکال از واحد ضرب است. ما مقداری را در ضرب‌کننده (مضروب فیه) واحد ضرب قرار می‌دادیم و بعد که به آن رجوع می‌کردیم دیگر آنجا نبود. ما مشکل را به شرکت DEC (دیجیتال اکوئیمپنت) گزارش کردیم و آن‌ها نتوانستند علتش را بیابند و دلشان هم نمی‌خواست که به آن بپردازند. در آن روزها نمودارهای مدار^{۴۳} ماشین‌ها را به خریدار سخت‌افزار می‌دادند و نهایتاً ما خودمان خطا را از طریق مطالعه نمودار مدار کشف کردیم. بعد به شرکت DEC تلفن کردیم و گفتیم «آن سیم را به آن سیم وصل کنید.» سیپل: خوشبختانه سخت‌افزارها این روزها چنین مشکلاتی بار نمی‌آورند.

تامپسون: بله، چنین اشکالاتی بسیار نادرند. به علاوه، آن موقع ما به زبان همگذاری (اسمبلی) برنامه می‌نوشتیم و در آن زبان به راحتی امکان داشت که در فراخوانی یک زیرروال^{۴۴}، مقدار اشتباهی را در یک ثبات^{۴۵} داشته باشید. اما اکنون که از زبان‌های سطح بالا استفاده می‌شود تمام نشانوندها^{۴۶} باید انطباق داشته باشند و در نتیجه، این‌گونه خطاها واقعاً به ندرت ممکن است پیش بیایند. اگر خطا نرم‌افزاری باشد معمولاً در یک نقطه از برنامه اتفاق می‌افتد و معمولاً هم خطا با چیز دیگری ارتباط دارد. و شما می‌توانید به پایش اجرای برنامه بپردازید و هر کجا خطا روی داد به سرعت اجرا را قطع کنید و به ردیابی علت بروز خطا بپردازید.

اما خطایی که من از آن صحبت کردم چنین نبود و تنها پس از آن که من آن برنامه را که به ضرب و تقسیم زیادی نیاز داشت نوشتم، بسامد تکرارش زیاد شد. به جای آن که هر چند روز یکبار اتفاق افتد، اینک هر چند دقیقه یکبار اتفاق می‌افتاد.

سیپل: امروز بعضی‌ها می‌گویند «خوب، مطمئناً زبان همگذاری (اسمبلی) این‌گونه فرصت‌ها را برای خراب کردن محتوای حافظه از طریق خطاهای نرم‌افزاری به وجود می‌آورد، اما زبان C نیز از بسیاری از زبان‌های دیگر استعداد بیشتری برای تولید خطا دارد.» اشاره‌گرها می‌توانند بسیار گیج‌کننده و مولد خطا باشند و شما می‌توانید از انتهای آرایه^{۴۷} عبور کنید. این‌ها به نظر شما مسأله‌زا نیست؟

تامپسون: نه. بعضی افراد کدهای ضعیف و شکننده‌ای می‌نویسند و برخی دیگر کدهای بسیار خوب و ساختیافته. و این به خود افراد برمی‌گردد. به نظر من، تقریباً در هر زبانی شما می‌توانید کد ضعیف و شکننده‌ای بنویسید. تعریف من از کد شکننده این است که فرض

کنید بخواهید ویژگی تازه‌ای را به یک برنامه بیفزایید. در یک کد خوب تنها یک نقطه وجود دارد که شما باید آن ویژگی را در آنجا اضافه کنید و جا می‌افتد. اما در یک کد شکننده شما برای این کار باید به ده جای برنامه دست بزنید و تغییرات ایجاد کنید.

سیپل: پس وقتی که رخنه‌گران^{۴۸} با سوء استفاده از سرریز میانگیر^{۴۹} مشکل امنیتی به وجود می‌آورند، پاسخ شما به کسانی که C و ++C را نیز تا حدی مسئول این مشکل می‌دانند چیست؟ آن‌ها می‌گویند اگر از زبان دیگری استفاده شده بود که محدوده آرایه را کنترل می‌کرد یا زباله‌روبی^{۵۰} داشت، از بسیاری از این‌گونه خطاها جلوگیری می‌شد.

تامپسون: خطا خطاست، چه فرقی می‌کند. ممکن است زبانی اجازه سرریز میانگیر را ندهد اما در عوض سیستم‌عامل فروپاشد^{۵۱}.

سیپل: اما فرق است بین حمله انکار سرویس^{۵۲} و به دست آوردن ریشه^{۵۳} و انجام هر کار دلخواه.

تامپسون: اما سرریز میانگیر تنها راه به دست آوردن ریشه نیست. راه دیگرش این است که برنامه را وادار کنید کاری که نباید انجام دهد را انجام بدهد. و این راه بسیار متداول‌تر از سرریز میانگیر است. بنابراین استدلال شما درست نیست.

سیپل: بسیار خوب، مسأله سرریز میانگر و به دست آوردن ریشه را کنار می‌گذاریم. رده‌ای از خطاها هستند که در C و ++C به دلیل مشابهی اتفاق می‌افتند که مثلاً در جاوا اتفاق نمی‌افتند. بنابراین آیا برای انواع خاصی از کاربردها، آیا مزیتی که از اجازه دادن به آن رده از خطاها به دست می‌آید به مشکلاتی که به وجود می‌آورند می‌ارزد؟ تامپسون: من فکر می‌کنم آن رده از خطاها بخش کوچکی از مشکلات را تشکیل می‌دهند.

سیپل: شما به هنگام اشکال زدایی از چه ابزارهایی استفاده می‌کنید؟ تامپسون: من غالباً از دستورات چاپ استفاده می‌کنم. هنگامی که در حال نوشتن برنامه هستم، حجم عظیمی خروجی چاپ شده تولید می‌کنم. من به ندرت مجبور می‌شوم که برای اشکال زدایی دوباره به برنامه بازگشت کنم.

سیپل: چه چیزهایی را چاپ می‌کنید؟

تامپسون: هر چیزی که نیاز دارم، مثلاً مقادیر ثابت^{۵۴} حلقه‌ها. اما غالباً از دستورات چاپ هنگامی که در حال نوشتن برنامه هستم استفاده می‌کنم نه پس از آن. این شیوه اشکال‌زدایی من است. من برنامه‌ها را از نقطه صفر شروع نمی‌کنم. برنامه‌ای را می‌گیرم و اصلاحش می‌کنم. حتی یک برنامه بزرگ را. از خودم می‌پرسم نخستین چیزی که می‌خواهم چاپ شود چیست و ابتدا آن قسمت را

48- hackers

49- buffer overflow

50- garbage collection

51- crash

52- denial-of-service

53- root

54- invariants

43- circuit diagram

44- subroutine

45- register

46- argument

47- array

در ۹۹ درصد موارد، چیزی که ساده و سراسر است نوشته شده باشد، درست کار می‌کند. اگر برنامه‌ای که نوشته‌اید مورد استفاده وسیعی قرار می‌گیرد و رتبه آماریش^{۵۹} از n^2 بیشتر است در این صورت باید به فکر بهینه‌سازی باشید. اما معمولاً این‌گونه نیست. هر چه ساده‌تر، بهتر.

سیبل: اما بعضی‌ها کاری به زمان اجرای برنامه ندارند بلکه کمال‌گرا هستند و می‌خواهند کدشان مثل جواهر بدرخشد.

تامپسون: خوب، من هم همین‌طور هستم. اما بخشی از آن مربوط به قربانی کردن الگوریتم در برابر کد است. منظورم این است که یک الگوریتم پیچیده نوعاً متضمن کد پیچیده‌ای است. و من الگوریتم ساده و کد ساده را بسیار بیشتر ترجیح می‌دهم. و اگر تنها یک چیز مشخصه کد من باشد، سادگی آن است. هیچ چیز عجیب و غریبی ندارد و همه به راحتی می‌توانند آن را بخوانند.

سیبل: آیا هنوز کارهایی هست که مردم بخواهند برای بهبود زمان اجرای آن‌ها به زبان همگذاری (اسمبلی) روی آورند؟

تامپسون: به ندرت. خیلی به ندرت، مگر این که مربوط به بخش‌های خاصی باشد. اگر شما یک برنامه مترجم بنویسید، مطمئناً ۹۹ درصد کدی که تولید می‌کنید یک یا دوبار اجرا خواهد شد. اما بخشی از آن در داخل سیستم‌عامل قرار می‌گیرد که بیست و چهار ساعته اجرا می‌شود و بخشی از آن در داخل داخلی‌ترین حلقه‌های سیستم‌عامل. بنابراین، اگر ۰/۱ درصد در این بخش از برنامه مترجم بهینه‌سازی صورت گیرد، تأثیر عمیقی بر عملکرد سیستم می‌گذارد. پس در این‌گونه موارد ممکن است به زبان همگذاری روی آورید.

سیبل: اما این می‌تواند در نتیجه تولید کد بهتر توسط برنامه مترجم هم به دست آید تا نوشتن برنامه مترجم به زبان همگذاری.

تامپسون: اوه، بله. حتماً.

سیبل: و شاید یک دلیل این که این روزها نوشتن برنامه‌ها مستقیماً به زبان همگذاری اهمیت کمتری یافته است این باشد که برنامه‌های مترجم بهتر شده‌اند.

تامپسون: نه. من فکر می‌کنم عمدتاً به خاطر این است که ماشین‌ها بسیار بهتر شده‌اند. برنامه‌های مترجم اصلاً خوب نیستند. شما به کدی که از GCC بیرون می‌آید نگاه کنید. افتضاح است. واقعاً خیلی بد است. و خیلی هم کند است. خود برنامه مترجم بیش از ۲۰ گذر دارد. به نحو وحشتناکی کند است اما رایانه‌ها از زمانی که GCC تولید شده تاکنون ۱۰۰۰ بار سریع‌تر شده‌اند. بنابراین ممکن است به نظر آید که سریع‌تر شده اما این به خاطر سرعت رایانه‌ای است که روی آن اجرا می‌شود.

سیبل: موضوع دیگری که تا حدی به بحث ما مربوط است، زیالهرویی است. نظرتان در این باره چیست؟ مثلاً GC در جاوا کم و بیش جا افتاده است. یکبار دنیس ریچی گفته بود که C به طور جدی مخالف زیالهرویی است. آیا به نظر شما حرکت به سمت چنین زبان‌هایی

می‌نویسم و اشکال زدایی می‌کنم. من در خلال تولید نرم‌افزار، ساعتی ۲۰ بار برنامه را اجرا می‌کنم و تکه‌ها را بر روی هم می‌سازم.

سیبل: شما گفتید که مقادیر ثابت حلقه‌ها را چاپ می‌کنید. آیا از احکام اظهاری^{۵۵} هم برای واریسی مقادیر ثابت استفاده می‌کنید؟

تامپسون: به ندرت. من اطمینان حاصل می‌کنم که برنامه درست است و سپس یا چیزهایی که چاپ شده را به عنوان توضیحات^{۵۶} به برنامه اضافه می‌کنم یا آن‌ها را دور می‌ریزم.

سیبل: چرا برای شما ساده‌تر است که مقادیر ثابت را چاپ کنید و خودتان آن‌ها را چشمی بررسی کنید تا این که از جملات اظهاری استفاده کنید و بگذارید به طور خودکار واریسی شوند؟

تامپسون: زیرا هنگامی که شما چیزی را چاپ می‌کنید به طور واقعی می‌بینید که چیست و به علاوه، من بجز مقادیر ثابت، خیلی چیزهای دیگر را هم چاپ می‌کنم. به هر حال، این شیوه کار من است. نمی‌خواهم آن را به عنوان یک الگو معرفی کنم.

سیبل: شما هنگامی که درباره طراحی نرم‌افزار صحبت کردید، آن را به صورت یک فرایند پایین به بالا توصیف کردید. آیا شما این قطعات پایین به بالا را به طور انفرادی می‌سازید؟

تامپسون: گاهی اوقات بله.

سیبل: و برای آزمایش این توابع سطح پایین آیا خودتان آزمایش^{۵۷} می‌نویسید؟

تامپسون: بله، اغلب اوقات خودم می‌نویسم. البته به برنامه‌ای که می‌نویسم هم بستگی دارد. اگر برنامه مبتدل A به B باشد، در این صورت من برنامه را با تمام A اجرا می‌کنم و جواب را با B‌های مربوط مقایسه می‌کنم. اما نوع دیگری از برنامه‌ها هستند که این‌گونه نیستند. البته من خیلی درگیر آزمایش نرم‌افزار نبوده‌ام اما برای آزمایش این دسته از برنامه‌ها عمدتاً از آزمایش پسنمایی^{۵۸} (آزمایش دوباره کارکرد کل سیستم پس از اتصال هر مؤلفه جدید به منظور اطمینان از این که مؤلفه جدید عملکرد سیستم را خراب نکرده باشد. مترجم) استفاده می‌کنم.

سیبل: یک مرحله دیگر برنامه‌نویسی، بهینه‌سازی است. برخی افراد بهینه‌سازی را از همان اول انجام می‌دهند. برخی دیگر ترجیح می‌دهند ابتدا نسخه عملیاتی برنامه را تولید کنند و بعداً به بهینه‌سازی آن بپردازند. رویکرد شما چیست؟

تامپسون: من در همان بار اول به ساده‌ترین شکل برنامه را می‌نویسم و در اغلب موارد برای همیشه همان‌طور باقی می‌ماند. نوشتن الگوریتم بسیار پیچیده‌ای برای چیزی که هیچوقت اجرا نمی‌شود، کار احمقانه‌ای است. تلف کردن وقت است. مؤلف خطاست. و نگهداریش غیرممکن است زیرا باید ۵۰ صفحه ریاضیات به کار ببرید تا به یک نفر دیگر بگوئید که چه کار کرده‌اید.

55- assertion

56- comment

57- test case

58- regression test

59- order statistics

خوب است؟ آیا این فناوری استحقاق این که نهایتاً مورد استفاده همگانی قرار گیرد را دارد؟

تامپسون: جوابش را نمی‌دانم. من درباره این موضوع دچار دوگانگی هستم. اگر شما دارید یک سیستم عامل یا مترجم زبان C یا چیزی که افراد بسیار زیادی از آن استفاده می‌کنند می‌نویسید، به عقیده من زباله‌روبی تقریباً اشتباه است. نوعی تقلب است زیرا شما می‌توانید به طور دستی آن را بسیار بهتر انجام دهید. کاری که می‌کنید این است که کار و وظیفه خودتان را به دوش چیز دیگری می‌اندازید و آن را برای کاربران کندتر می‌کنید. بنابراین به نظر من در سیستم عامل اشتباه است. اما اگر دارید برنامه‌ای می‌نویسید که کاری را برایتان انجام دهد و بعداً آن را دور می‌اندازید، در این حالت بد نیست. یک بخش از کارهایی که نمی‌خواهید به آن بپردازید را از پیش رویتان بر می‌دارد و به خاطر سرعت رایانه‌ها، هزینه‌اش هم برایتان قابل قبول است. در نتیجه، همان‌طور که گفتیم واقعاً در این مورد دچار دوگانگی هستیم.

یک مسأله دیگر هم این است که الگوریتم‌های زباله‌روبی متفاوتی وجود دارند با خصوصیات بسیار متفاوت از یکدیگر. بنابراین اگر دارید یک کار همه منظوره انجام می‌دهید و نمی‌دانید که کاربران واقعیتان چه کسانی هستند به مشکل بر می‌خورید. هیچ الگوریتم زباله‌روبی که برای همه ماشین‌ها مناسب باشد وجود ندارد.

سیپل: شما خودتان را دانشمند می‌دانید یا مهندس یا هنرمند یا صنعتگر یا یک چیز دیگر؟

تامپسون: نمی‌دانم. من از به کار بردن واژه دانشمند بیزارم زیرا فکر می‌کنم نخبه‌گرایانه است. و بر درجه دکتری دلالت دارد. هیچ گواهی و مدرکی وجود ندارد که بگوید شما پس از گذراندن این دوره‌ها یا قبولی در این درس‌ها «دانشمند» شده‌اید. بنابراین من از این واژه خوشم نمی‌آید و آن را به کار نمی‌برم. مهندس، شاید. من مدرکی دارم که رویش نوشته شده «مهندسی». پس می‌توانم از واژه «مهندس» استفاده کنم. و هنگامی که در جایی مشغول کار می‌شدم، عنوان شغلم را مهندس یا برنامه‌نویس می‌گذارم زیرا برایش توجیه دارم. اما غالباً این عناوین برای من اهمیتی ندارد و به آن‌ها توجه نمی‌کنم.

سیپل: صرف‌نظر از این که خودتان را چه خطاب می‌کنید، شما با کدامیک از آدم‌های زیر شباهت بیشتری حس می‌کنید فیزیکدان، کسی که پل می‌سازد، نقاش یا نجار؟

تامپسون: رده‌های پایین‌تر. صنعتگر با کمی هم ذوق هنری.

سیپل: شما برنامه‌نویسان مستعد را چگونه شناسایی می‌کنید؟

تامپسون: کار سرگرم کننده‌ای است. از آن‌ها می‌پرسم که جالب‌ترین برنامه‌ای که روی آن کار کرده‌اند چه بوده است. و آن‌ها را وادار می‌کنم که درباره الگوریتم‌هایش و ساختمان داده‌هایش و معماریش توضیح دهند. اگر نتوانستند در مقابل پرسش‌های من مقاومت کنند، در این صورت خوب نیستند. اگر بتوانم به آن‌ها حمله کنم یا مشکلاتی در الگوریتم‌ها و راه‌حل‌هایشان پیدا کنم و آن‌ها با وجودی که خیلی بیشتر از من درگیر بوده‌اند نتوانند به خوبی دفاع کنند، نه،

به درد نمی‌خورند. این شیوه مصاحبه من است. به من گفته‌اند که مصاحبه‌هایم خیلی سخت و کوبنده هستند.

سیپل: می‌توانم تصوّر کنم. مثل یک امتحان شفاهی است. آیا فکر نمی‌کنید به آدم‌هایی برخوردید که مستقل از قابلیت‌های برنامه‌نویسی‌شان، از نظر شخصیتی نتوانند با این گونه مصاحبه‌ها کنار بیایند؟

تامپسون: نه، فکر نمی‌کنم مستقل از برنامه‌نویسی باشد. اگر شروع کنم از آن‌ها سؤالات درسی بپرسم ممکن است، اما من که این کار را نمی‌کنم. من از آن‌ها می‌خواهم کاری که انجام داده‌اند و وقت زیادی برایش صرف کرده‌اند را توضیح دهند. من هرگز کسی را ندیده‌ام که روی کاری واقعاً وقت گذاشته باشد و مشتاق نباشد که توضیح دهد چکار کرده، چگونه کرده و چرا آن کار را کرده است. انتخاب موضوع به عهده خود آن‌هاست. من که برای آن‌ها موضوع انتخاب نمی‌کنم. بنابراین من در آن موضوع آماتور هستم و آن‌ها حرفه‌ای. اگر آن‌ها نتوانند تحمل کنند که یک آماتور از آن‌ها درباره حرفه‌شان سؤال کند، معلوم می‌شود که یک جای کار اشکال دارد.

سیپل: شما اکنون در گوگل چکار می‌کنید؟

تامپسون: کارهای زیرساختی. مثل سیستم عامل. چسباندن تکه‌های مختلف به هم. من اجازه دارم که هر کاری دلم خواست انجام دهم. چالشی که وجود دارد این است که کاری کنیم که تعدادی ماشین غیرقابل اطمینان مثل یک ماشین چند پردازنده‌ای⁶⁰ قابل اطمینان کار کنند.

سیپل: چهارچوب⁶¹ معروف MapReduce گوگل هم برای همین کار به وجود آمده؟ (MapReduce نام چند چهارچوب نرم‌افزاری است. یکی از آن‌ها در سال ۲۰۰۴ توسط گوگل معرفی شده و هدفش پشتیبانی از رایش توزیعی⁶² بر روی مجموعه‌های بزرگ داده‌ها بر روی خوشه‌هایی⁶³ از رایانه‌هاست. مترجم)

تامپسون: خوب، فرایندی است که قواعد معنایی⁶⁴ معروفی دارد. و حلقه‌های بازخوردی⁶⁵ ندارد. اگر ساختار قابل اطمینانی برای انجام آن داشته باشید خیلی از مسائل را می‌توانید در آن ساختار جای دهید.

سیپل: و شما هم بر روی چیزهایی در همین چهارچوب کار می‌کنید؟ **تامپسون:** نه. من سعی می‌کنم بارسنگین قابلیت اطمینان را از دوش برنامه‌نویسان منفرد بردارم. چالش بسیار سختی است. تمام نرم‌افزارها در اینجا پر هستند از لایه‌های اگر این کار نکند چه اتفاقی می‌افتد، اگر آن کار نکند چه اتفاقی می‌افتد. من فکر می‌کنم ۵۰ درصد کدها از این نوع what-if هاست.

سیپل: پس هدف شما کنار گذاشتن آن ۵۰ درصد است؟

تامپسون: امیدوارم. کار سختی است.

سیپل: از کار کردن در گوگل راضی هستید؟

60- multiprocessor

61- framework

62- distributed computing

63- cluster

64- semantics

65- feedback loop

رسیدم که گفتم دیگر پس است. بیش از این با آن کار نخواهم کرد. من در مصاحبه‌ای دقیقاً گفتم که از ++C استفاده نکردم زیرا برای دو روز متوالی یک جور نمی‌ماند. وقتی استراستروپ مصاحبه‌ام را خواند با داد و فریاد به اتاق من آمد و گفت که من دارم زیر پایش را خالی می‌کنم و حرف‌های من تأثیرگذار است و گفت که من گفته‌ام ++C زبان بدی است. در حالی که من هرگز نگفته بودم که زبان بدی است. این جر و بحث مدتی ادامه داشت و پس از آن من از وارد شدن به این گونه بحث‌ها دوری می‌کنم.

سیبیل: اکنون می‌توانید بگوئید که به نظر شما زبان بدی است یا زبان خوبی است؟

تامپسون: مطمئناً نکته‌های خوب و مثبتی هم دارد. اما به طور کلی فکر می‌کنم زبان بدی است. خیلی چیزها را نیمه کاره انجام داده و یک خروار ایده‌های دو به دو ناسازگار را روی هم ریخته است. هر کس که من می‌شناسم، چه به طور فردی و چه شرکتی، زیر مجموعه‌ای از آن را انتخاب کرده و به کار می‌برد و این زیر مجموعه‌ها با هم متفاوتند. بنابراین، زبان خوبی برای انتقال الگوریتم نیست. به نوعی خیلی بزرگ و پیچیده است. و به وسیله یک کمیته ساخته شده است. استراستروپ سال‌ها و سال‌ها فعالیت کرد، خیلی فراتر از مشارکت فنی که در خود زبان داشت، تا این زبان را جا بیندازد و برایش کاربر پیدا کند. او تقریباً اداره تمام کمیته‌های استاندارد مربوط را به دست گرفت و به هیچکس «نه» نگفت. او هر ویژگی و امکاناتی که وجود داشت را در ++C گنجاند. ++C طراحی شسته رفته‌ای نداشته است. صرفاً اجتماع^{۶۶} هر چیزی است که به ذهن رسیده. و من فکر می‌کنم که از این بابت به شدت لطمه خورده است. **سیبیل:** آیا فکر می‌کنید که این به خاطر این بوده که او تمام ایده‌ها را دوست داشته یا این که می‌خواست بدین ترتیب به جا انداختن زبان کمک کند و به هر کس هر چه می‌خواهد بدهد؟

تامپسون: فکر می‌کنم بیشتر مورد دوم بوده است.

سیبیل: خیلی‌ها هستند که می‌گویند «++C مزخرف است». با وجود این، همه از آن استفاده می‌کنند. مثلاً یکی از چهار زبان رسمی گوگل است. اگر این قدر که شما می‌گویید بد است، پس چرا هنوز به استفاده از آن ادامه می‌دهند؟

تامپسون: نمی‌دانم. فکر می‌کنم علاقه به ++C در گوگل در حال فروکش کردن است. اکنون تعداد کسانی که آن را دوست ندارند بیشتر از طرفدارانش است.

سیبیل: و آن‌ها به جاوا روی می‌آورند؟

تامپسون: نمی‌دانم. تقریباً جایگزینی برای ++C وجود ندارد. همه از آن گله دارند اما سراغ زبان دیگری نمی‌روند. فارغ‌التحصیلان دانشگاهی - یعنی همان‌هایی که گوگل استخدامشان می‌کند - آن را بلدند. در نتیجه خیلی سخت است که کار دیگری بکنند. دلیلی که ادامه می‌دهند همین است. صرفه‌جویی در هزینه‌های

تامپسون: بخشی از آن را خیلی دوست دارم. اما بخشی دیگر خیلی سنگین و حجیم است زیرا همه چیز با پول سر و کار دارد. اندازه‌ها غیرقابل تصورند. امروز یک نرم‌افزار می‌نویسید و فردا دو میلیون کاربر پیدا می‌کند. تصورش را هم نمی‌توانید بکنید. مسئولیت‌ها بسیار سنگین هستند.

سیبیل: و شما در بخش تولید هستید، به جای آن که در آزمایشگاه‌های گوگل باشید که با سابقه شما در آزمایشگاه‌های بل همخوانی بیشتری دارد.

تامپسون: در واقع من در بخش تولید هم نیستم. من در پروژه‌هایی هستم که به تولید می‌انجامد. احتمالاً شرح وظیفه من پیدا کردن روشی برای بهتر کردن زندگی است (این که من واقعاً وظیفه‌ام را انجام می‌دهم یا نه، پرسش دیگری است). یا ارزیابی ایده‌های جدید کارمندان برای جایگزین ساختن چیزهای قدیمی. سعی می‌کنم آن‌ها را بهتر کنم. به طور خلاصه، وظیفه من این است که سعی کنم هر چیزی در ساختار گوگل که می‌تواند بهتر انجام گیرد را بهتر کنم.

سیبیل: من می‌دانم که گوگل سیاستی دارد مبنی بر این که هر کارمند جدید پیش از این که اجازه داشته باشد به بررسی و کنترل برنامه‌های نوشته شده به یک زبان بپردازد، باید از نظر تسلط بر آن زبان مورد امتحان و بررسی واقع شود. معنی این حرف این است که شما هم از نظر تسلط بر زبان C مورد امتحان قرار گرفته‌اید.

تامپسون: بله، این سیاست گوگل است اما من مورد امتحان قرار نگرفته‌ام.

سیبیل: یعنی شما اجازه ندارید که کد C را بررسی و ارزیابی کنید؟ **تامپسون:** نه، اجازه ندارم.

سیبیل: آیا این نشانه نوعی اعتراض فلسفی شما به استاندارد برنامه‌نویسی گوگل است؟

تامپسون: من مورد امتحان قرار نگرفته‌ام و تاکنون نیازی به این کار پیش نیامده است.

سیبیل: پس شما کارهایتان را درون قصر شیشه‌ای خودتان انجام می‌دهید. آیا اغلب برنامه‌هایتان را به C می‌نویسید؟

تامپسون: من غالباً به زبان C برنامه‌نویسی می‌کنم. در حالی که در گوگل همه چیز به ++C است. من مشکلی در برنامه‌نویسی به زبان ++C ندارم اما از آن خوشم نمی‌آید. و در مقابلش مقاومت می‌کنم.

سیبیل: شما با بیارنه استراستروپ (خالق زبان ++C، مترجم در AT&T مالک آزمایشگاه‌های بل، مترجم) بودید. آیا هیچ در تولید ++C مشارکت داشتید؟

تامپسون: اگر پاسختان را بدهم به دردسر می‌افتم!

سیبیل: اشکالی ندارد.

تامپسون: من در زمانی که ++C در حال تولید بود با آن کار کردم و نظراتم را در موردش ابراز داشتم. این بخشی از جو کاری آنجا بود. من یک تکه برنامه می‌نوشتم و فردایش دیگر کار نمی‌کرد زیرا زبان تغییر یافته بود. برای مدتی طولانی خیلی ناپایدار بود. بالاخره به نقطه‌ای

گزارش باز آموزی. با حفظ ++C آدم‌ها زودتر به مرحله تولید می‌رسند.

سیپل: چه زبان یا زبان‌های دیگری هستند که شما از آن‌ها خوشتان می‌آید یا از برنامه‌نویسی به آن‌ها لذت برده‌اید؟

تامپسون: من تقریباً تمام زبان‌های خاص را یکبار امتحان کرده‌ام. مثل Maple و Macsyma برای حل معادلات و SNOBOL برای کار با رشته‌ها. من با یک دوجین زبان که چیز جالبی داشته‌اند کار کرده‌ام.

سیپل: از چه ابزارهای تولید نرم‌افزار خوشتان می‌آید؟

تامپسون: من عاشق Yacc هستم. درست همان کاری را می‌کند که شما می‌خواهید. مکمل آن، یعنی Lex، وحشتناک است. هیچ کاری که شما می‌خواهید را انجام نمی‌دهد.

سیپل: آیا به هر حال از آن استفاده می‌کنید یا کارهای مربوط به lex را دستی انجام می‌دهید؟

تامپسون: دستی می‌نویسم. خیلی راحت‌تر است.

سیپل: آیا تاکنون برنامه‌نویسی ادیبانه^{۶۷} دونالد کنوث را انجام داده‌اید؟

تامپسون: نه. ایده فوق‌العاده‌ای است اما تقریباً انجام دادن آن در عمل غیرممکن است.

سیپل: چرا؟

تامپسون: دو باز نمود یک برنامه است که غالباً با یکدیگر تناقض دارند. و هیچ راهی برای رفع آن وجود ندارد. اگر چیزی به خوبی به یک زبان برنامه‌نویسی نوشته شده باشد، در این صورت خواناست. و همین کفایت می‌کند. نیازی به توضیحات موازی نیست. توضیحات شاید برای الگوریتم‌ها بد نباشد و یا اگر می‌خواهید ترفندی به کار برید به صورت هشدار یا چیزی شبیه آن، من از طرفداران استفاده زیاد از جملات توضیحی نیستم.

سیپل: در مصاحبه‌ای که با کنوث داشتم، او گفت کلید فنی نویسی^{۶۸} این است که هر چیز را دوبار بگوئید و به شکل مکمل هم. بنابراین، من فکر می‌کنم او این مسأله را به عنوان یک ویژگی برای برنامه‌نویسی ادیبانه قلمداد می‌کند، نه یک اشکال.

تامپسون: خوب، اگر دو راه برای بیان داشته باشید یکی از آن‌ها واقعی است: آنچه ماشین اجرا می‌کند. و دیگری چنین نیست. تنها هنگامی که یکی از آن‌ها خیلی خلاصه‌تر و کوتاه‌تر از دیگری باشد ارزش خواندنش را دارد. اگر هر دو یک حجم داشته باشند، شما آن که کار می‌کند را می‌خوانید. اگر یکی از آن‌ها خیلی خلاصه‌تر اما کمتر دقیق باشد و شما بتوانید هر چه نیاز دارید را از آن استخراج کنید عالیست. اما اگر نتوانید، سراغ آن دیگری می‌روید. بسته به این که دنبال چه چیزی باشید، یکی از آن دو را می‌خوانید. اما اگر منظور این است که یک الگوریتم را تا جزئی‌ترین سطح، یکبار به زبان برنامه‌نویسی و یکبار به انگلیسی، بنویسیم شاید کنوث بتواند این کار را بکند اما من نمی‌توانم.

67- literate programming

68- technical writing

سیپل: آیا تاکنون هیچ یک از برنامه‌های ادیبانه او را خوانده‌اید؟
تامپسون: فقط مطالبش در مقاله‌های اولیه را خوانده‌ام. به تازگی چیزی نخوانده‌ام.

سیپل: آیا کتاب‌هایی هستند که به نظر شما مهم باشند؟ منظورم این است که یا برای خودتان مهم بوده‌اند یا خواندنشان را به دیگران توصیه می‌کنید؟

تامپسون: من کتاب‌های مقدماتی برنامه‌نویسی را نمی‌خوانم. بنابراین، در توصیه چنین کتاب‌هایی مشکل دارم. اگر بخواهم زبان جدیدی یاد بگیرم، سعی می‌کنم کتاب مناسبی پیدا کنم. من معمولاً کتاب‌های عمیق‌تر و پیچیده‌تر که قواعد نحوی و معنایی را به من یاد دهد ترجیح می‌دهم تا کتاب‌هایی که می‌گویند سبک خوب چیست و سبک بد چیست.

هنگامی که می‌خواستم تدریس کنم تمام کتاب‌های درسی موجود را می‌خواندم تا از بینشان انتخاب کنم. اما به غیر از آن، من چنین کتاب‌هایی نمی‌خوانم.

سیپل: در داستان ابداع یونیکس توسط شما خوانده‌ایم که همسر و فرزندان برای یک ماه به مسافرت رفتند و شما در آن یک ماه، کار به این بزرگی انجام دادید. قاعدتاً باید ساعت‌های طولانی در آن ماه صرف کار کردن کرده باشید. ما چرا این‌گونه کار می‌کنیم؟ آیا لازم است؟ و یا فقط به خاطر این که سرگرم کننده است چنین می‌کنیم؟
تامپسون: این کارها انگیزه لازم دارد. انگیزه‌ای که شما را بی‌اختیار به انجام دادنش بکشاند. من فکر نمی‌کنم که می‌توانستم انجامش دهم. نکته دیگر این که وقتی همسر و فرزندان در کنارتان باشند، شما یک چرخه ۲۴ ساعته منظم در شبانه روز دارید. وقتی آن‌ها به مسافرت رفتند من دیگر چرخه ۲۴ ساعته نداشتم. و من معمولاً ۶ ساعت می‌خوابیدم و ۲۷ یا ۲۸ ساعت کار می‌کردم. به علاوه، هنگامی که آدم به طور عادی از خواب بیدار می‌شود حالش برای کار کردن بهتر از موقعی است که با صدای گریه بچه از خواب می‌پرد.

سیپل: این مربوط به موقعی است که انگیزه انجام یک پروژه شما را به پیش می‌راند و شما به محض آن که از خواب بیدار می‌شوید می‌خواهید هر چه زودتر خود را به رایانه برسانید و شروع به نوشتن کدهای بیشتر بکنید. اما دلیل دیگری که آدم‌ها ساعت‌های طولانی کار می‌کنند این است که باید پروژه‌ای را در یک مهلت زمانی مقرر تمام کنند و این باعث می‌شود ۸۰ تا ۱۰۰ ساعت در هفته کار کنند.
تامپسون: این کار آدم را از پا می‌اندازد. اما آن کاری که من کردم هرگز برایم استرس‌زا نبود. من اسم این نوع کار را برنامه‌نویسی هیجان‌انگیز می‌گذارم. من در موقعیت‌هایی که مهلت‌های زمانی تولید استرس می‌کنند نیز قرار داشته‌ام. دیگر جنبه سرگرمی کار از بین می‌رود. من آن را دوست ندارم.

سیپل: آدم در آخر کار از پا می‌افتد که این بد است اما از لحاظ انجام کار در مدتی کوتاه، آیا جواب می‌دهد؟

تامپسون: معمولاً شما در وضعیتی قرار می‌گیرید که این کار

سیبل: خوب این ساز و کاری برای رایانش توزیعی است.
تامپسون: بله، اینجا این کار را می‌کنند اما خواندن و درک کردنش بسیار سخت است.
سیبل: هنگامی در یک تیم کار می‌کنید، چه ساختاری را دوست دارید؟

تامپسون: فقط می‌توانم بگویم کار کردن با افراد خوب و هماهنگ.
سیبل: هنگامی که با آدم‌های جور و هماهنگ کار می‌کنید آیا بیشتر طرفدار مالکیت شخصی کد هستید، یعنی بگوئید «من این تکه از کد را نوشته‌ام، مال من است و مسئولیتش با من است» یا مالکیت اشتراکی: «همه ما با هم مالک این کد هستیم و هر کداممان می‌تواند هر قسمت را که تشخیص داد اصلاح کند»؟
تامپسون: من همیشه در بین این دو سبک کار کرده‌ام. یک نفر مسئولیتش را بر عهده دارد و اگر شما مشکلی با آن داشتید به او می‌گویید و وظیفه اوست که مشکل را برطرف کند. و در بعضی موارد هم آن‌ها ناپدید می‌شوند، نمی‌خواهند کد را تغییر دهند، پاسخگو نیستند- در این حالت خودتان اصلاح می‌کنید. بنابراین، بین این دو حالت است. کار اصولی این است که همیشه تقاضاها اول برای کسی که کد را نوشته ارسال شوند.

سیبل: این روزها بعضی‌ها طرفدار برنامه‌نویسی دو نفره^{۶۹} هستند، یعنی دو نفر با هم پشت یک صفحه کلید کار کنند. آیا شما هیچگاه بدین شکل کار کرده‌اید؟

تامپسون: کارهای کوچک را می‌توان بدین شکل انجام داد. گاهی اوقات پیش آمده که من داشتم تایپ می‌کردم و یک نفر دیگر که دستش خیلی از من تندتر بوده، جای من نشسته و من خوانده‌ام و او تایپ کرده است. یکی دو بار هم با یک نفر دیگر مشترکاً کاری را که هر کدام به تنهایی می‌توانستیم انجام دهیم، با هم انجام داده‌ایم.
سیبل: نتیجه کار، بهتر یا سریعتر نشد؟

تامپسون: نتیجه کار بهتر نبود. شاید اشکال زدایی سریعتر شد زیرا وقتی شما داشتید تایپ می‌کردید یک نفر دیگر از پشت سرتان می‌توانست خطاها را تشخیص دهد. در نتیجه، فکر می‌کنم تعداد خطاها کمتر از حد معمول شد. اما این کار به عنوان یک فلسفه یا روش کار به نظرم نیامد.

سیبل: آیا شما هنوز از برنامه‌نویسی لذت می‌برید؟
تامپسون: بله. من برنامه‌های کوچک را دوست دارم. منظورم از کوچک این است که بتوان ظرف یک ماه آن را نوشت. اما اگر کاری باشد که مثلاً یک سال طول بکشد، من دیگر حوصله‌اش را ندارم.
سیبل: آیا همیشه این گونه بوده یا شما دیگر انرژی لازم برای پروژه‌های بزرگتر را ندارید؟

تامپسون: نمی‌دانم. بستگی به کارهایی که به طور واقعی پیش می‌آیند دارد. یک کار بزرگ که سال‌ها به طول می‌انجامد، مثل یک سیستم‌عامل، شما آن را به بخش‌هایی تقسیم می‌کنید و خیلی

ادامه‌دار می‌شود. به محض آن که یک مهلت تمام شد، یکی دیگر در افق پدیدار می‌گردد. اگر شما به طور ثابت تحت فشار مهلت‌های زمانی پروژه‌ها باشید، در این صورت کم‌کم اشتیاق و انگیزه‌تان را از دست می‌دهید و واقعاً نمی‌توانید به این شیوه زندگی ادامه دهید. من که نمی‌توانم.

سیبل: تحت فشار مهلت‌های زمانی قرار داشتن با تخمین نادرست زمان انجام پروژه نیز ارتباط دارد. آیا شما می‌توانید تخمین بزنید نوشتن یک برنامه چقدر زمان می‌گیرد؟

تامپسون: بستگی به این دارد که برنامه را برای خودم می‌نویسم یا برای استفاده دیگران. اگر برنامه را دارم برای خودم می‌نویسم زمانش را می‌توانم تخمین بزنم. من می‌توانم با چیزهای غیر منتظره آن کنار بیایم. من معمولاً دنبال انجام «آن ۱۰ درصد اضافی دیگر» نمی‌روم. من می‌توانم از تله‌هایی که می‌دانم در آن وجود دارد دوری کنم. چیزهایی مثل این. من می‌توانم برنامه را آماده کنم و بعد تر و تمیزش کنم و در این مدت از آن استفاده کنم. شاید این تعریف متفاوتی از «پایان کار» باشد. اما اگر بخواهم برنامه را برای استفاده دیگران بنویسم، معمولاً افراد دیگری نیز دخیل می‌شوند که باید با آن‌ها هماهنگ بود. در این صورت، نمی‌توانم زمان تکمیل برنامه را تخمین بزنم.

سیبل: شما در مصاحبه‌ای در سال ۱۹۹۹ گفتید که به لینوکس زیاد فکر نمی‌کنید. اکنون پس از گذشت ده سال چه فکری در مورد آن می‌کنید؟ حالا که دارد دنیا را فرا می‌گیرد.

تامپسون: اکنون خیلی قابل اطمینان‌تر شده است. شکی در آن نیست. من گاه‌گاه به کدش نگاه می‌کنم. البته نه مثل سابق. هنگامی که روی سیستم‌عامل Plan 9 کار می‌کردم، برای نوشتن گرداننده‌های^{۷۰} سخت‌افزارهای مختلف، به گرداننده‌های لینوکس نگاه می‌کردم. اکنون نیازی به نگاه کردن به کد لینوکس ندارم. من لینوکس را اجرا می‌کنم و گاهی هم به کدش نگاه می‌کنم اما به ندرت. بنابراین واقعاً نمی‌توانم بگویم که کیفیتش بهتر شده است یا نه. اما مطمئناً قابلیت اطمینانش بهتر شده است.

سیبل: آیا هیچگاه صرفاً برای سرگرمی کد می‌خوانید؟
تامپسون: در گذشته این کار را می‌کردم اما اکنون کمتر.
سیبل: آیا برنامه‌ای را می‌گرفتید و کاملاً آن را درک می‌کردید یا فقط نگاهی به آن می‌انداختید تا ببینید ساختارش چگونه است و بعضی کارها را چگونه انجام داده است؟

تامپسون: کمی از هر دو. من در ابتدا روال‌های کتابخانه‌ای را انتخاب می‌کردم. در مورد بعضی برنامه‌ها فقط به برنامه اصلی^{۷۱} نگاه می‌کردم. اینجا سبک برنامه‌نویسی در گوگل بسیار عجیب و باور نکردنی است. آن‌ها یک فراخوانی زیر روال^{۷۱} را می‌گیرند، آن را به صورت یک RPC بسته‌بندی می‌کنند و در جایی به صورت ایستا ذخیره می‌کنند. این یعنی هر کس می‌تواند در هر زمان و به هر دلیل آن را فراخوانی کند.

69- drivers

70- main program

71- subroutine call

72- pair programming

تکه‌های سرگرم کننده و لذت بخش در آن وجود دارد. بنابراین می‌توانید آن را چند چیز کوچک در نظر بگیرید نه یک چیز بزرگ. اما خیلی چیزها هم هستند که فقط یک چیز بزرگند و من همیشه با این گونه پروژه‌ها مشکل داشته‌ام. من به بازخورد و خشنودی ناشی از دیدن نتیجه کار نیاز دارم. اما اگر مجبور باشید بنشینید و روزها و ماه‌ها کار کنید و کار کنید و کار کنید و هیچ چیز بجز حجم زیادی کد ننسید، من با این نوع کار کردن مشکل دارم.

سیبیل: شما غالباً در کارهای پژوهشی مشارکت داشته‌اید و به نظر می‌رسد که آزادی عمل زیادی برای کار کردن در زمینه‌های مورد علاقه خود داشته‌اید. آیا وقتی کار شما به یک شغل تبدیل شد، این روند تغییر کرد؟ آیا جنبه‌های سرگرم کننده و لذت بخش کار را کمتر کرده است؟

تامپسون: نه. همیشه سرگرم کننده بوده است و علتش هم عمدتاً این است که من کارهایی را که دوست داشته‌ام انتخاب کرده‌ام. و حتی زمانی که دنبال شغل بودم، یعنی دوران دانشکده، هزاران شغل موجود بود و به نظرم رسید که هزاران نفر که مشغول کارند، هر کاری باشد، به کارهای کوچک برنامه‌نویسی برای کمک به کار اصلی‌شان نیازمندند. و این برای من عالی بود. کارهای کوچک برنامه‌نویسی که می‌توانستم ظرف چند روز انجام و تحویل دهم و از بینشان کار مورد علاقه‌ام را انتخاب کنم.

فکر می‌کنم اولین کاری که گرفتم از یک استاد دانشکده انسان‌شناسی بود که می‌خواست کار هومر را فهرست‌بندی کند. او ایلید و اودیسه را بر روی کارت منگنه داشت و می‌خواست بسامد واژه‌های به کار رفته در این دو اثر را به دست آورد. یعنی یک نوع تحلیل آماری این دو اثر. و برای من خیلی سرگرم کننده بود. کار متن‌پردازی^۳ بود که آن روزها توسط رایانه‌ها انجام نمی‌گرفت. این نخستین کاری بود که برایش پول گرفتم.

سیبیل: شما در مصاحبه سال ۱۹۹۹ گفتید که به پسران توصیه کرده‌اید به جای دانش رایانه، زیست‌شناسی بخواند زیرا فکر می‌کردید در دنیای رایانه‌ها چیز جدیدی وجود ندارد. این مربوط به تقریباً ۱۰ سال پیش است. اکنون درباره آن چه حسی دارید؟

تامپسون: احساس مشابهی دارم. چیز چندان جدیدی که نمی‌توانستید پیش‌بینی کنید در دنیای رایانه‌ها اتفاق نیفتاده است. آخرین چیز مهم به نظر من اینترنت بود که در سال ۹۹ هم وجود داشت. همه چیز فقط توسعه یافته است. سرعت رایانه‌ها هنوز هم به طور نمایی در حال افزایش است، اما چه فرقی می‌کند؟ **سیبیل:** تاریخچه یونیکس را که می‌خوانیم به نظر می‌رسد شما و دنیس ریچی آن سیستم‌عامل را به این خاطر ابداع کردید که می‌خواستید با یک رایانه خاص کار کنید. منظورم این است که در آن زمان، برای انجام کارهایی که امروزه خیلی ساده و ابتدایی به نظر می‌رسند مثل نوشتن یک بازی یا چیزی بر روی رایانه، شما

مجبور بودید کل سیستم‌عامل را بنویسید. شما مجبور بودید که برنامه‌های مترجم را بنویسید و بسیاری از زیرساخت‌ها را فراهم کنید تا قادر به انجام کاری باشید. مطمئنم که همه آن کارها برای خودشان سرگرم کننده بوده‌اند. اما پیچیدگی برنامه‌نویسی مدرن که پیش از این صحبتش را کردیم، با تمام این لایه‌های چسبیده به هم، حداقل باعث شده که دیگر مجبور به انجام آن کارها نباشیم. دیگر نخستین گام این نیست که برای خودتان سیستم‌عامل بنویسید.

تامپسون: بله، اما بدتر از آن است. سیستم‌عامل نه تنها به شما داده می‌شود، بلکه انتخاب نوعش نیز اجباری است. اگر با کسانی که این روزها از دانشکده‌های علوم و مهندسی رایانه بیرون می‌آیند مصاحبه کنید، متوجه می‌شوید که مطلقاً شالوده و ساخت‌های زیربنایی رایانش را درک نمی‌کنند. دور بودن آن‌ها از این که رایانه چیست و حتی نظریه رایانش، واقعاً ترسناک است. آن‌ها این چیزها را درک نمی‌کنند.

سیبیل: داشتیم به نصیحت شما به پسران که گفتید رشته زیست‌شناسی را به جای رایانه انتخاب کند فکر می‌کردم. آیا فکر نمی‌کنید خود برنامه‌نویسی، مستقل از این که برنامه‌نویس چقدر دور یا نزدیک به سخت‌افزاری که با آن کار می‌کند باشد، به عنوان یک سرگرمی ذهنی به قدر کافی جذابیت داشته باشد؟ تعریف یک فرایند که توسط این ماشین‌های معجزه‌گر برایتان کاری انجام دهد.

تامپسون: من فکر می‌کنم تغییر کرده است. شاید به دلیل بالا رفتن ستم باشد اما به نظرم می‌رسد هنگامی که شما لایه دیگری را روی لایه دیگری بنا می‌کنید، دیگر از مزایای برنامه‌نویسی، آن گونه که شما گفتید، بهره‌مند نمی‌شوید. من فکر می‌کنم الگوریتم‌ها در طول زمان پیچیده‌تر شده‌اند. یک الگوریتم جدید برای انجام یک کار، بر اساس ۵۰ الگوریتم کوچک دیگر بنا شده است. موقعی که ما بچه بودیم این الگوریتم‌های کوچک را می‌نوشتیم که بسیار سرگرم کننده و جذاب بود. ما آن‌ها را کاملاً درک می‌کردیم. اما امروز کاری که می‌کنید این است که مسأله را به بخش‌هایی تقسیم می‌کنید و می‌دانید که هر بخش با فلان الگوریتم حل می‌شود، بدون آن که الگوریتم را عمیقاً درک کرده باشید و بدانید چگونه کار می‌کند. بنابراین، فکر می‌کنم برنامه‌نویسی امروز با سابق فرق دارد. من واقعاً به این مسأله اعتقاد دارم و علتش این است که همه چیز در طول زمان لایه‌بندی شده و ما اکنون فقط با لایه‌ها سروکار داریم. شاید هم درک لایه‌ها به مذاق من خوش نمی‌آید.

«ادامه دارد...»

منبع:

* "Coders at Work", Peter Seibel, Apress, 2009

مروری بر روش‌های جستجو در داده‌های رمز شده

لیلا معظمی گودرزی

مجتمع دانشگاهی فناوری اطلاعات، ارتباطات و امنیت، دانشگاه صنعتی مالک اشتر

پست الکترونیکی: lm8001@gmail.com

مجید غیوری

عضو هیأت علمی دانشگاه امام حسین(ع)، دانشکده علوم و فنون هوایی

پست الکترونیکی: ghayoori@gmail.com

چکیده

برون‌سپاری داده‌ها با وجود مزایای فراوان، دارای مشکلاتی مربوط به امنیت داده‌ها می‌باشد. لذا ناگزیر باید داده‌های حساس رمز شوند که در نتیجه آن مسئله مدیریت داده‌های رمز شده مطرح می‌شود که شامل مسایلی چون مدیریت کلید، صحت داده‌ها و پرس و جو روی داده‌های رمز شده است. در این نوشته قصد داریم انواع روش‌های مطرح شده برای جستجو روی داده‌های رمز شده را بیان کنیم. دسته‌بندی این روش‌ها از نظر این که روی چه نوع داده‌ای اعمال شوند به سه دسته داده‌های رمز شده در قالب‌های رابطه‌ای، متنی و XML تقسیم می‌شوند که ما به بررسی چند روش جستجو در اسناد XML پرداخته‌ایم. از نظر نوع روش نیز به دسته‌های زیادی چون استفاده از شاخص، توابع هم‌ریخت اختفایی، اشتراک راز، توابع حفظ ترتیب و ... تقسیم می‌شوند.

کلمات کلیدی

جستجو در داده‌های رمز شده، XML، شاخص ساختاری

۱- مقدمه

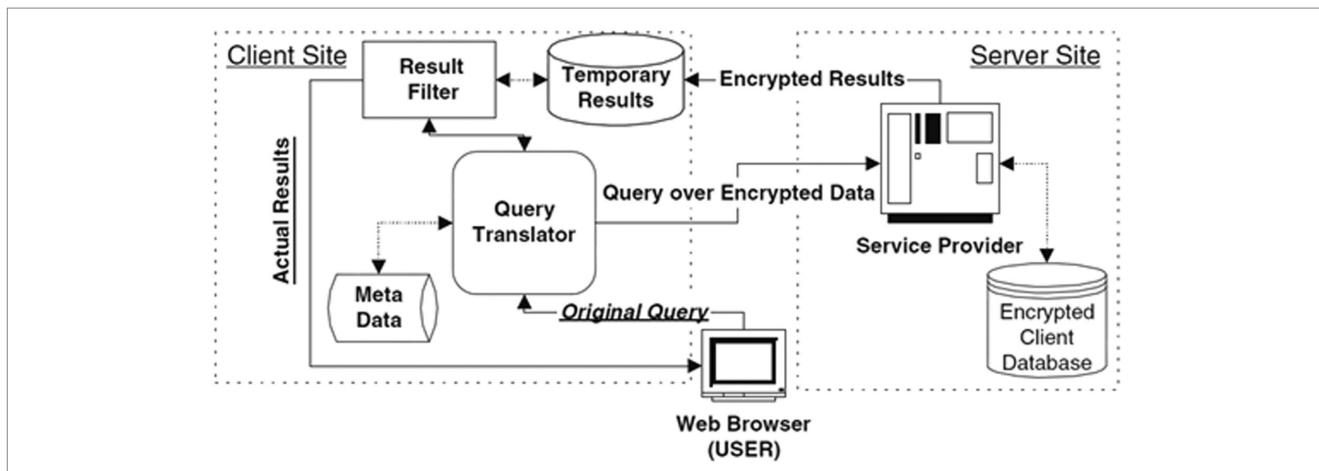
اخیراً در تجارت گرایش به سوی تمرکز بر کار اصلی و برون‌سپاری سایر کارهاست. شرکت‌ها نرم‌افزارها و پایگاه داده‌های خود را برون‌سپاری

می‌کنند. شرکت‌ها با استفاده از ¹DAS وظایف مدیریت پایگاه داده مانند پشتیبان‌گیری، بازگرداندن اطلاعات، دسترس‌پذیری و مدیریت فضا را برون‌سپاری می‌کنند. DAS باعث داشتن ذخیره قابل اطمینانی از حجم زیاد داده و پردازش کارای پرس و جو می‌شود. از طرفی در صورت اطمینان نداشتن به کارساز، مسایل امنیتی ایجاد خواهد شد.

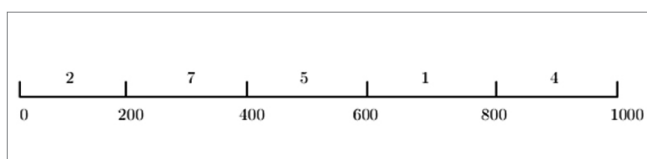
همان‌طور که می‌دانیم کنترل دسترسی برای محافظت از داده‌ها کافی نیست، چرا که امکان سرقت رسانه محتوی داده وجود دارد. همچنین عدم اعتماد به اعمال کننده کنترل دسترسی یکی دیگر از عواملی است که سبب می‌شود برای حفظ محرمانگی داده‌ها در کارگزار غیر قابل اعتماد، از رمزنگاری استفاده شود که در نتیجه آن مسئله پرس و جو روی داده‌های رمز شده مطرح می‌شود.

روش اصلی پردازش پرس و جوهای رمز شده فرستادن کل پایگاه داده رمز شده به مالک داده است. اما این روش سبب ترافیک بیهوده در کانال ارتباطی، رمزگشایی و پردازش پرس و جو در سمت مشتری می‌شود و کارایی خوبی ندارد. برای سرعت بخشیدن به پردازش پرس

1-Database As a Service



شکل (۱) معماری عمومی پردازش جست و جو در داده‌های رمز شده



شکل (۳) شاخص‌دهی با استفاده از باکت‌بندی

eid	ename	salary	addr	did
23	Tom	70K	Maple	40
860	Mary	60K	Main	80
320	John	50K	River	50
875	Jerry	55K	Hopewell	110

شکل (۲) جدول با مقادیر اصلی

می‌شود و در کنار داده رمز شده در کارساز ذخیره می‌شود تا برای جستجو استفاده شود. شاخص نباید موجب فاش شدن یا استنتاج قسمتی از داده شود. ایجاد شاخص به روش‌های گوناگون صورت می‌گیرد: ایجاد شاخص با استفاده از باکت‌بندی، تابع درهم‌ساز^۲ و درخت B+. توضیح هر کدام از این روش‌ها در ادامه خواهند آمد.

۲-۱- باکت‌بندی

در این روش دامنه مقادیر هر صفت به تعدادی بازه افراز می‌شود. سپس به هر بازه یک مقدار نسبت داده می‌شود، این مقدار می‌تواند با یک تابع شبه تصادفی به دست آید. (شکل ۳) در نهایت این شاخص در ستون مربوطه صفت در کارساز ذخیره می‌شود. همچنین رمز شده هر سطر نیز نگهداری می‌شود. (شکل ۴)

پرس و جوهای تساوی با استفاده از این نوع شاخص امکان‌پذیر است. پرس و جوی بازه‌ای نیز با توجه به نوع بازه‌بندی تا حدی امکان‌پذیر است و قابل تبدیل به چند پرس و جوی تساوی است. البته اشتباهات زیادی در جواب به وجود می‌آید که باید در سمت مشتری اصلاح شود. امکان اجرای توابع تجمعی MIN و MAX در این نوع شاخص وجود دارد. اما نمی‌توان SUM و AVG را بدون رمزگشایی داده‌ها اجرا کرد.

بسته به چگونگی انجام افراز، ممکن است کارساز بتواند در مورد

و جو باید بیشتر بارکاری در فراهم‌کننده سرویس باشد که معمولاً توانایی‌های بیشتری از مشتری دارد.

در بخش‌های بعد، به بیان انواع روش‌های گفته شده برای پرس و جو روی داده‌های رمز شده می‌پردازیم. نکته مهم در همه این روش‌ها این است که از انواع پرس و جوها پشتیبانی می‌کنند یا خیر. انواع پرس و جوها شامل پرس و جوهای تساوی، بازه‌ای و تجمعی برای داده‌های عددی و پرس و جوهای تطبیق دقیق و الگویی برای داده‌های نویسه‌ای است هرچند ما در این گفتار فقط به بررسی پرس و جو روی داده‌های عددی می‌پردازیم.

معماری عمومی پردازش پرس و جوی رمز شده در شکل ۱ نشان داده شده است که روند آن به صورت زیر می‌باشد:

کاربر یک پرس و جو ایجاد می‌کند. این پرس و جو با استفاده از میدل پرس و جوی سمت مشتری به شکل رمز شده تبدیل می‌شود و برای کارساز ارسال می‌شود. کارساز یک مجموعه پاسخ به مشتری برمی‌گرداند که معمولاً پاسخ دقیق نیست. مشتری نتایج را رمزگشایی می‌کند و آن‌ها را فیلتر می‌کند. پس مشتری باید توانایی پردازش آخر نتایج را داشته باشد. در بعضی معماری‌ها لازم است چند پیام بین مشتری و کارساز تبادل شود تا نتیجه به دست آید. [۱]

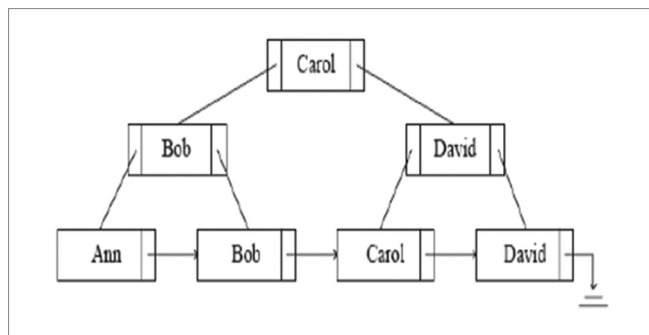
۲- استفاده از شاخص

شاخص فراداده‌ای است که با استفاده از داده اصلی ساخته

2-hash function

<i>etuple</i>	<i>eid^S</i>	<i>ename^S</i>	<i>salary^S</i>	<i>addr^S</i>	<i>did^S</i>
1100110011110010...	2	19	81	18	2
1000000000011101...	4	31	59	41	4
1111101000010001...	7	7	7	22	2
1010101010111110...	4	71	49	22	4

شکل (۴) جدول با مقادیر شاخص



شکل (۵) درخت B+

<u>Id</u>	<u>VertexContent</u>	<u>Id</u>	<u>C</u>
1	2, Carol, 3	1	gtem945/*c
2	4, Bob, 5	2	8dq59wq*d'
3	6, David, 7	3	ue63/)jw
4	Ann, 5, 1, 5	4	8/*5sym,p
5	Bob, 6, 2, 3	5	mw39wio[
6	Carol, 7, 4	6	=wco21!ps
7	David, NIL, 6	7	oieb5(p8*

شکل (۶) جدول بالا: جدول درخت B+، جدول پایین: شاخص رمز شده

می‌دهد. رابطه برای هر راس درخت یک ردیف دارد و اشاره‌گرها از طریق ارجاع متقابل از محتوای راس به شناسه راس‌های دیگر در رابطه نمایش داده می‌شوند.

از آنجا که درخت شامل اطلاعات حساس است، رمزنگاری در سطح راس اعمال شده است (چندتایی در رابطه). رابطه رمز شده متناظر دو صفت دارد: Id که شناسه راس است و C که شامل راس رمز شده است. (شکل ۶)

شاخص‌گذاری B+ اجازه انجام پرس و جوی بازه‌ای، تساوی، تجمعی، ترتیبی و گروه‌بندی شده را می‌دهد.

برای مثال برای اجرای پرس و جوی $a_{ij} \geq V$ که V یک مقدار ثابت است، درخت B+ باید پیمایش شود تا برگ یا برگ‌های درست پیدا شوند. ابتدا مشتری پرس و جویی برای بازبایی ریشه درخت مطرح می‌کند. مثلاً اگر Id ریشه ۱ باشد او چندتایی t را می‌خواهد که $t[Id]=1$. $t[C]$ را رمزگشایی کرده و محتوایش را با V مقایسه می‌کند، اگر $K_{i+1} < V \leq K_i$ ، اشاره‌گر P_i انتخاب می‌شود. این روند ادامه می‌یابد تا یک برگ پیدا شود و پس از آن کل زنجیره برگ‌ها طی می‌شود تا تمام چندتایی‌های $a_{ij} \geq V$ استخراج شوند. این روش در جواب پرس و جو چندتایی اضافه بر نمی‌گرداند اما گرانتر

داده‌ها استنتاج انجام دهد. برای جلوگیری از این کار، تعداد داده‌ها در هر بازه نباید خیلی کم یا خیلی زیاد باشد. یک بحث مهم در ایجاد این نوع شاخص، به هنگام‌سازی افزاز است که با تغییر توزیع داده‌ها نیاز است انجام شود. [۷][۸]

۲-۲- توابع درهم‌ساز

شاخص در این نوع با استفاده از توابع درهم‌ساز ساخته می‌شود. مقدار شاخص، خروجی تابع درهم‌ساز مقدار صفت است. از آنجا که برد توابع درهم‌ساز کوچکتر از دامنه آن‌هاست ممکن است خروجی آن‌ها دارای برخورد^۳ باشد و در نتیجه ممکن است مقدار شاخص دو مقدار متفاوت، یک مقدار یکسان شود. لذا در جواب پرس و جوی اشتباه وجود دارد و مشتری باید آن را پس از رمزگشایی اصلاح کند اما این ویژگی موجب می‌شود که کارساز نتواند با استفاده از نتایج پرس و جوی استنتاج انجام دهد.

چون تابع درهم‌ساز ترتیب مقادیر دامنه را حفظ نمی‌کند، فقط پرس و جوی‌های تساوی در این نوع شاخص امکان‌پذیر است و امکان انجام پرس و جوی بازه‌ای و تجمعی در آن وجود ندارد. [۷][۵]

۲-۳- درخت B+

یک درخت B+ با خروجی n یک درخت است که هر راس آن می‌تواند تا $n-1$ مقدار کلیدی جستجو و n اشاره‌گر را ذخیره کند. بجز ریشه و برگ‌ها هر راس حداقل $\lceil n/2 \rceil$ فرزند دارد. یک راس درونی، f مقدار کلیدی $K_1, \dots, K_f$ را با $f \leq n-1$ ذخیره می‌کند. هر مقدار کلیدی K_i یک اشاره‌گر P_i دارد. P_0 به زیردرختی اشاره می‌کند که شامل کلیدهایی با مقادیر کمتر از K_1 است، P_f به زیر درختی اشاره می‌کند که شامل مقادیری بزرگتر یا مساوی K_f است و هر P_i به زیردرختی با مقادیر $[K_i, K_{i+1})$ اشاره می‌کند. راس‌های داخلی به طور مستقیم به چندتایی‌های^۴ پایگاه داده اشاره نمی‌کنند فقط به راس‌های دیگر اشاره می‌کنند. برعکس راس‌های برگ به چیزی اشاره نمی‌کنند بلکه به طور مستقیم به چندتایی‌های پایگاه داده ارجاع می‌دهند که یک مقدار مشخص برای صفت شاخص‌گذاری شده دارند. برگ‌ها در یک زنجیره متصل هستند که اجازه اجرای کارای پرس و جوی بازه‌ای را می‌دهد. (شکل ۵)

یک درخت B+ می‌تواند برای هر صفت a_{ij} در شمای رابطه R_i گرفته شود که a_{ij} روی یک دامنه ترتیب جزیی تعریف شده است. شاخص توسط مالک داده روی مقادیر صفت ساخته می‌شود و همراه با پایگاه داده رمز شده روی کارگزار ذخیره می‌شود. برای این منظور، ساختار B+ به یک رابطه خاص با دو صفت تبدیل می‌شود: Id که شناسه راس را نشان می‌دهد و vertex content که محتوای واقعی راس را نشان

3-collision

4-tuple

با پایگاه داده رمز شده روی کار ساز ذخیره می شود. متفاوت از ساختار شاخص درخت B+، ساختار داده درخت B که در OPESS استفاده شده روی مقادیر شاخص ساخته شده است و نه روی مقادیر متن ساده. بنابراین قبل از ساختن ساختار درخت B که روی کار گزار ذخیره می شود، OPESS دو تکنیک روی مقادیر اصلی a_{ij} اعمال می کند: تقسیم و مقیاس بندی که یک توزیع فرکانسی مسطح را روی مقادیر شاخص به دست می آورند. [۵]

صفت a_{ij} را که روی دامنه D_{ij} تعریف شده را در نظر بگیرید و فرض کنید مقادیر $\{V_1, \dots, V_n\}$ در رابطه r_i به ترتیب به تعداد $\{f_1, \dots, f_n\}$ رخ داده اند. ابتدا یک فرایند تقسیم روی a_{ij} انجام می شود که تعدادی شاخص که توزیع فرکانسی مسطحی دارند ایجاد می کند. فرایند تقسیم به هر مقدار V_h از a_{ij} در π اعمال می شود. سه عدد صحیح مثبت متوالی $m-1$ ، m ، $m+1$ تعیین می شوند به طوری که فرکانس f_h به صورت ترکیب خطی زیر نمایش داده می شود:

$$f_h = C_1(m-1) + C_2(m) + C_3(m+1)$$

C_1 ، C_2 و C_3 مقادیر صحیح غیر منفی اند.

مقدار رمز نشده V_h می تواند به مقادیر شاخص C_1 با $m-1$ تکرار، مقادیر شاخص C_2 با m تکرار و مقادیر شاخص C_3 با $m+1$ تکرار نگاشت شود. برای مثال عدد ۹۰ که ۳۴ بار تکرار شده است:

$$34 = 1*6 + 4*7 + 0*8$$

عدد ۹۰ به $1+4+0=5$ مقدار متفاوت با ۵ کلید متفاوت رمز می شود که تعداد فرکانس ۶ یا ۷ دارند.

توجه شود که همواره سه گانه $(m-1, m, m+1)$ موجود است چرا که بجز مقادیر با فرکانس ۱، سه گانه $(۲, ۳, ۴)$ برای همه می تواند استفاده شود. سعی می شود بیشترین m انتخاب شود تا کمترین کلید به دست آید چرا که کلیدها باید ذخیره شوند. آن مقادیری که یکبار تکرار شده اند را به m مقدار می شکنیم.

در نهایت بیشترین تعداد کلید مورد نیاز عبارت است از:

$$K = \max_{1 \leq i \leq n} \{C_1^i + C_2^i + C_3^i\}$$

برای حفظ ترتیب مقادیر شاخص با توجه به دامنه اصلی صفت a_{ij} ، برای هر دو مقدار $V_h < V_l$ و مقادیر شاخص i_h و i_l مرتبط با آن ها باید داشته باشیم: $i_h < i_l$. برای به دست آوردن مقادیر تقسیم شده V_i به صورت زیر عمل می کنیم:

K عدد تصادفی $w_1, \dots, w_k \in (0, 1/K+1)$ را انتخاب می کنیم. بدون از دست دادن کلیت فرض می کنیم:

$$w_1 < w_2 < \dots < w_k, V_1 < V_2 < \dots < V_k$$

تعریف می کنیم: $\delta = \max_{1 \leq i \leq n-1} (V_{i+1} - V_i)$

اولین تکه V_i را برابر $enc(V_i + w_1 \delta)$ می گیریم و n امین آن را $enc(V_i + (\sum_{j=1}^n w_j) \delta)$ در نظر می گیریم. نتایج با استفاده از یک تابع رمزنگاری حفظ ترتیب رمز می شوند.

مثال: فرض کنیم $K=4$ ، $m=6$ و $\delta=9$ باشد و عدد ۱۰۰۱، ۲۳ بار تکرار شده است. داریم:

از روش های باکت و توابع درهم ساز است چرا که برای هر پرس و جو به اندازه ارتفاع درخت نیاز به تبادل اطلاعات بین کارخواه و کار گزار می باشد. برای بهبود می توان یکی از روش های باکت یا تابع درهم ساز را با روش شاخص درخت ترکیب کرد. دقت شود که چون محتوای درخت B+ رمز شده است در برابر حملات استنتاج امن است. [۵]

۳- روش های با حفظ ترتیب

در روش رمزنگاری با حفظ ترتیب^۵ یا OPES داده ها به گونه ای رمز می شوند که ترتیب داده ها پس از رمز شدن با ترتیب داده های اصلی یکسان باشد. این نوع رمزنگاری بدون استفاده از درخت B+ برای پرس و جوهای بازه ای مناسب است و جواب اشتباه نیز بر نمی گرداند. OPES فقط در صورتی محرمانگی داده را تامین می کند که مهاجم اطلاعی از داده های اصلی یا دامنه صفات اصلی نداشته باشد.

روش کلی توابع حفظ ترتیب سه مرحله است:

- مدل کردن: داده ها در این مرحله دسته بندی می شوند و دسته ها در مراحل رشد و هرس کوچک و بزرگ می شوند تا دسته بندی مناسب به دست آید.
- مسطح کردن: داده های به دست آمده از مرحله اول در این مرحله به یک توزیع یکنواخت نگاشت می شوند.
- تبدیل: داده های به دست آمده به یک توزیع مقصد دلخواه کاربر نگاشت شده و رمز می شوند.

در تمام این مراحل، ترتیب حفظ می شود.

بر اساس OPES یک روش دیگر به نام ROPES پیشنهاد شده که امنیت OPES را افزایش داده است. به این صورت که بین مرحله دوم و سوم یک مرحله تصادفی کردن اضافه می شود که موجب می شود نگاشت بین دسته ها در توزیع مبدا و مقصد را تصادفی کند. یعنی ترتیب داده ها در هر دسته حفظ می شود اما دسته ها به صورت تصادفی به هم می ریزند. این روش هر چند در برابر حمله متن اصلی معلوم مقاوم است اما اجرای پرس و جوهای بازه ای در آن سخت تر است.

OPESS^۶ تکامل OPES است که پرس و جوهای بازه ای را پشتیبانی می کند و مشکل استنتاج هم ندارد. این روش شاخص گذاری از ساختار درخت B سنتی استفاده می کند که توسط پایگاه داده رابطه ای برای شاخص دهی فیزیکی داده استفاده می شود. ساختار داده درخت B شبیه به B+ است اما راس های داخلی به طور مستقیم به چندتایی ها در پایگاه داده ارجاع می دهند و برگ های درخت در یک لیست به هم وصل نشده اند.

یک شاخص OPESS می تواند برای هر صفت a_{ij} در شمای رابطه R_i گرفته شود که a_{ij} روی یک دامنه ترتیب جزیی تعریف شده است. شاخص توسط مالک داده روی مقادیر صفت ساخته می شود و همراه

5-Order Preserving Encryption Schema

6- Order Preserving Encryption with Splitting and Scaling

(a) <pre> <Bib> <Book> <Title>Spring</Title> <Author> <Name>F.WELL</Name> <Education> <BS>X School</BS> <Education> <Author> </Book> <Book> <Title>Football</Title> <Author> <Name>J.HAND</Name> <Education> <MS>X School</MS> <Education> <Author> </Book> </Bib> </pre>	(b) <table border="1"> <thead> <tr> <th>Node name</th> <th>LB</th> <th>RB</th> </tr> </thead> <tbody> <tr><td>Bib</td><td>1</td><td>26</td></tr> <tr><td>Book</td><td>2</td><td>13</td></tr> <tr><td>Title</td><td>3</td><td>4</td></tr> <tr><td>Author</td><td>5</td><td>12</td></tr> <tr><td>Name</td><td>6</td><td>7</td></tr> <tr><td>Education</td><td>8</td><td>11</td></tr> <tr><td>BS</td><td>9</td><td>10</td></tr> <tr><td>Book</td><td>14</td><td>25</td></tr> <tr><td>Title</td><td>15</td><td>16</td></tr> <tr><td>Author</td><td>17</td><td>24</td></tr> <tr><td>Name</td><td>18</td><td>19</td></tr> <tr><td>Education</td><td>20</td><td>23</td></tr> <tr><td>MS</td><td>21</td><td>22</td></tr> </tbody> </table> LB : Left Bound RB : Right Bound	Node name	LB	RB	Bib	1	26	Book	2	13	Title	3	4	Author	5	12	Name	6	7	Education	8	11	BS	9	10	Book	14	25	Title	15	16	Author	17	24	Name	18	19	Education	20	23	MS	21	22
Node name	LB	RB																																									
Bib	1	26																																									
Book	2	13																																									
Title	3	4																																									
Author	5	12																																									
Name	6	7																																									
Education	8	11																																									
BS	9	10																																									
Book	14	25																																									
Title	15	16																																									
Author	17	24																																									
Name	18	19																																									
Education	20	23																																									
MS	21	22																																									

شکل ۷) یک سند XML و شاخص آن

Function callInterval()

Input: Node p with interval $[min, max]$, weight w_1^1, w_2^1
 Output: Interval of p 's i th child c_i , $[min_i, max_i]$
 Assume p has N children.
 $d = (max - min) / (2 \times N + 1)$
 $min_i = min + (2 \times i - 1) \times d - d \times w_1^1$
 $max_i = min + 2 \times i \times d + w_2^1 \times d$

شکل ۸) الگوریتم ایجاد شاخص DSI

Node name	left Bound	right Bound
Bib	0	1
Book	0.12	0.56
Title	0.23	0.28
Author	0.34	0.54
...	...	...

شکل ۹) جدول شاخص DSI برای سند شکل ۸

Encryption Block Table		DSI Table	
ID	Interval	Tag	DSI
1	[0.23,0.28]	Bib	[0,1]
2	[0.34,0.54]	Book	[0.12,0.56]
		UXM45	[0.23,0.28]
		WRETS	[0.34,0.54]

شکل ۱۰) جدول DSI و جدول بلوک رمزنگاری

دو خاصیت جمعی و ضربی را پشتیبانی می‌کند، توابع جمع و ضرب را روی داده‌های رمز شده اجرا کرد. خواص تابع رمزنگاری هم‌ریخت استفاده شده به صورت زیر است:

- کلید رمزنگاری $K=(p, q)$ که p و q به وسیله کاربر تعیین می‌شوند.
- $n = p \cdot q$ که به کارگزار داده می‌شود.
- $qq^{-1} = 1 \pmod{p}$
- $pp^{-1} = 1 \pmod{q}$
- $E_k(a) = (a \pmod{p}, a \pmod{q})$
- $D_k(c1, c2) = (c1 \pmod{p}) qq^{-1} + (c2 \pmod{q}) pp^{-1} \pmod{n}$

$$23 = (2 \times 5) + (1 \times 6) + (1 \times 7)$$

$$w1 = 0.05, w2 = 0.1, w3 = 0.05, w4 = 0.02$$

$$1001 + 0.05 \times 9 = 1001.45$$

$$1001.45 + 0.1 \times 9 = 1002.35$$

$$1002.35 + 0.05 \times 9 = 1002.80$$

$$1002.80 + 0.02 \times 9 = 1002.98$$

این مقادیر رمز شده و به تعداد ۵، ۶ و ۷ بار تکرار می‌شوند.

چون در تکنیک تقسیم مجموع فرکانس شاخص‌های V برابر فرکانس اصلی V است، یک مهاجم که توزیع فرکانسی مقادیر رمز نشده دامنه را می‌داند می‌تواند از این ویژگی برای شکستن روش شاخص گذاری استفاده کند. در حقیقت مقادیر شاخص که به یک مقدار رمز نشده نگاشت شده‌اند مقادیر پیوسته هستند. برای این منظور پیشنهاد مقیاس بندی مطرح شد. هر مقدار رمز نشده V_h مرتبط با یک فاکتور مقیاس بندی S_h است. وقتی V_h به n مقدار شاخص به نام‌های $i_1, \dots, i_n$ تقسیم می‌شود. هر ورودی شاخص درخت B متناظر با i_j, S_h بار تکرار می‌شود. توجه شود که همه S_h نسخه تکراری از شاخص، به چندتایی‌های یکسانی در پایگاه داده رمز شده اشاره می‌کنند. S_h یک عدد حقیقی بین ۱ تا ۱۰ می‌باشد. بعد از اعمال مقیاس بندی توزیع فرکانسی شاخص، توزیع فرکانسی یکنواخت نیست. بدون دانستن فاکتور مقیاس بندی مهاجم نمی‌تواند ارتباط بین مقادیر شاخص و مقدار اصلی را بسازد.

این روش اجازه ارزیابی پرس و جوهای تساوی، بازه‌ای، ترتیبی، گروه بندی شده و بیشتر عملگرهای تجمعی را می‌دهد. اجرای پرس و جو در این روش گران است حتی اگر چندتایی‌های اضافه در جواب ایجاد نکند چرا که مقدار متن ساده یکسان به مقادیر شاخص متفاوت نگاشت شده‌اند و هر دو روش مقیاس بندی و تقسیم برای ارزیابی پرس و جو باید معکوس شوند. [۳]

۴- توابع هم‌ریخت اختفایی

در روش رمزنگاری هم‌ریخت اختفایی^۷، حاصل انجام یک عمل روی داده‌های رمز شده، معادل رمز شده حاصل عملی دیگر روی داده‌های اصلی است. مثلاً ضرب داده‌های رمز شده برابر رمز شده جمع اعداد است.

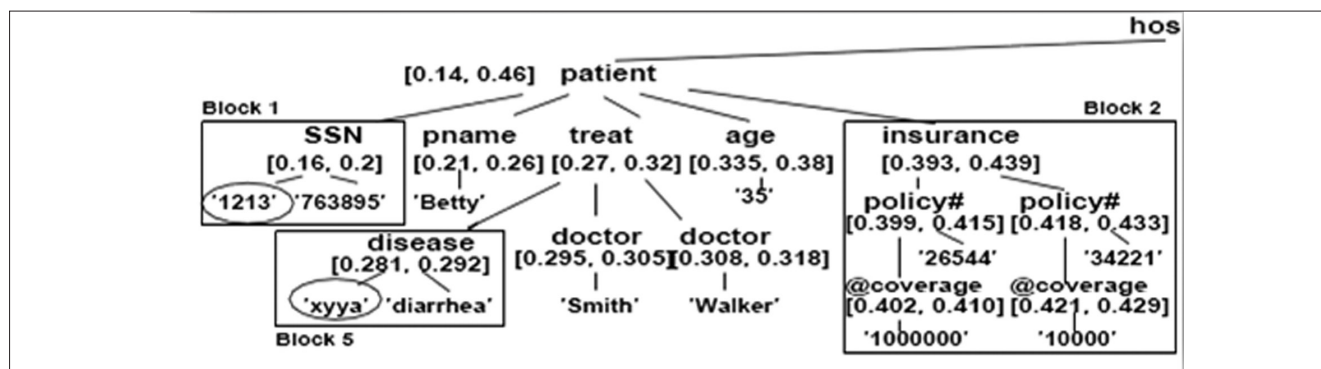
$$\beta(x^E, y^E) = (\alpha(x, y))^E$$

$$E(x) \cdot E(y) = E(x+y)$$

به وسیله توابع رمزنگاری هم‌ریخت اختفایی می‌توان برخی از عملیات مانند جمع و ضرب را به طور مستقیم روی داده‌های رمز شده انجام داد. حوزه‌هایی را که روی آن‌ها پرس و جوی تجمعی اجرا می‌شود، با استفاده از رمزنگاری هم‌ریخت رمز می‌کنند.

هاسیگموس با استفاده از یک تابع رمزنگاری هم‌ریخت اختفایی که

7-privacy homomorphism



شکل (۱۱) مثالی از بلوک‌های رمزنگاری

رمز شده است و ترکیب این مقادیر از نظر محاسباتی پیچیده است.

[۷]

$$p = 5, q = 7$$

$$n = p \cdot q = 35 \quad k = (5, 7)$$

با مثال زیر موضوع روشن تر می شود:

می خواهیم $a_1 = 8$ و $a_2 = 12$ را جمع کنیم.

$$E(a_1) = (3, 1)$$

$$E(a_2) = (2, 5)$$

$$E(a_1) + E(a_2) = (3+2, 1+5) = (5, 6)$$

$$D(5, 6) = 5 \cdot qq^{-1} + 6 \cdot pp^{-1} \pmod{35} = (5 \cdot 7 \cdot 3 + 6 \cdot 5 \cdot 3) \pmod{35} \\ = 195 \pmod{35} = 20$$

استفاده از تابع رمزنگاری هم‌ریخت اختفائی معرفی شده کاملاً امن نیست و کارساز می‌تواند مقدار اصلی برخی از مقادیر رمز شده را به دست آورد. برای امنیت بیشتر هاسیگموس نوفه^۴ تصادفی به مقادیر رمز شده اضافه می‌کند. $R(x)$ یک تابع شبه تصادفی است. نوفه در هنگام رمزگشایی از داده‌ها حذف می‌شود.

$$E(x) = (x \bmod p + R(x).p, x \bmod q + R(x).q)$$

اما اگر مقادیر تکراری در داده‌ها باشد نوفه نیز باعث حمله می‌شود. با وجود نوفه یک مقدار M تکراری به دو مقدار $C''(c'_1, c'_2)$ و $C''(c''_1, c''_2)$ نگاشت می‌شود.

$$c'_1 - c''_1 = R' \cdot p - R'' \cdot p = p \cdot (R' - R'')$$

چون R' و R'' کوچک هستند تجزیه $c'_1 - c''_1$ راحت است و p به راحتی به دست می‌آید. حتی اگر تجزیه $c'_1 - c''_1$ مشکل باشد محاسبه ب.م.م $c'_1 - c''_1$ و n ساده است که برابر با p می‌باشد. کافی است کارگزار برای اجرای این حمله بررسی فوق را روی هر جفت مقدار رمز شده برای یک صفت تا زمانی که جفت مقدار تکراری یافت شود انجام دهد. با یافتن p و q تمام مقادیر رمز شده برای کارگزار قابل رمزگشایی هستند. پس توابع هم‌ریخت اختفایی با استفاده از این روش ناامن است.

سیستم‌های دیگری که خاصیت جمعی را پشتیبانی می‌کنند ارائه شده‌اند که همه براساس سیستم رمزنگاری کلید عمومی با امنیت قابل اثبات هستند. مشکل این روش‌ها بزرگ بودن اندازه مقادیر

۵- جستجو در داده‌های XML رمز شده

جستجو در داده‌های XML رمز شده نیز مانند سایر مسایل قبلی است که گفته شد با این تفاوت که اینجا علاوه بر این که محتوای گره‌ها (مثل برچسب‌ها^۴ و مقادیر برگ‌ها) باید پنهان باشد ارتباط بین آن‌ها نیز باید مخفی بماند.

W3C استانداردهایی را برای رمزنگاری پیشنهاد داده است. طبق استانداردهای ذکر شده، برچسب‌ها و محتوای که قرار است رمز شوند با یک رشته به نام عنصر داده رمز شده جایگزین می‌شوند که زیرعنصر دارد:

- روش رمزنگاری: الگوریتم رمزنگاری و پارامترهای الگوریتم را نشان می‌دهد.

- اطلاعات کلید: نام کلید را بدون مقدارش نشان می‌دهد.

- داده رمز: عنصر رمز شده را همراه با محتوایش نشان می‌دهد.

- ویژگی‌های رمزنگاری: شامل اطلاعات اضافه مربوط به تولید داده رمز شده است.

اساساً دو نوع شاخص برای اسناد XML رمز شده استفاده شده است: شاخص ساختاری و شاخص مقداری. هدف شاخص ساختاری تعیین این است که مسیرها در پرس و جو با کدام مسیر در اسناد XML تطبیق داده شده است. هدف شاخص مقداری بررسی محدودیت‌ها در پرس و جوی بازه‌ای است. این شاخص‌ها می‌توانند سمت مشتری یا کارساز نگهداری شوند. [۲]

۵-۱- نگهداری شاخص‌ها سمت کارساز

یک شاخص ساختاری معروف در اسناد XML رمز نشده وجود دارد. در این ساختار شاخص به هر برچسب یک شماره دنباله داده شده است که از یک شروع می‌شود و یکی یکی زیاد می‌شود. هر برچسب دو شماره دنباله دارد که کران چپ و راست گره را نشان می‌دهند.

در مشتری پرس و جو را به شکل رمز شده تبدیل می کند. مبدل پرس و جو هر برجسب را با برجسب های رمز شده متناظر در شاخص ساختاری جایگزین می کند. DSI برجسب های موجود در پرس و جو از جدول DSI پیدا می شوند. این بازه ها برای فهمیدن id بلوک ها در جدول بلوک رمزنگاری استفاده می شوند. Id بلوک هایی که برگردانده می شوند نتایج پردازش شاخص ساختاری هستند. در مرحله دوم، مشتری محدودیت های مبتنی بر مقدار در پرس و جو را تبدیل می کند. هر محدودیت می تواند به صورت $\langle \text{tag}, \text{op}, \text{value} \rangle$ نوشته شود که برجسب و مقدار هر دو به صورت رمز شده قرار می گیرند. هر محدودیت مقداری به دلیل استفاده از تقسیم به یک پرس و جوی بازه ای روی شاخص درخت B تبدیل می شود. توجه کنید که این شاخص ها منطبق با بلوک های رمزنگاری هستند. کارساز id بلوکی را که شاخص مقداری را ارضا می کنند پیدا می کند. کارساز اشتراک id های بلوک از شاخص ساختاری و مقداری را می گیرد و بلوک ها از پایگاه داده رمز شده فراخوانی می شوند و نتیجه را برای رمزگشایی به مشتری می فرستد.

این روش قادر است به پرس و جوی های Min و Max پاسخ دهد اما نمی توان پرس و جوی های شمارشی و توابع جمعی را با استفاده از آن به دست آورد. هر چند شمارش را نمی تواند جواب دهد اما فرض بر این بوده که مهاجم فرکانس تکرار مقادیر دامنه را می داند و با این وجود این روش امن است.

مزایا:

- امکان انجام پرس و جوی بازه ای در سمت کارساز وجود دارد.
- شاخص های مقداری و ساختاری گفته شده امن قابل اثبات هستند.
- اطلاعات ساختاری حساس و ارتباط مقادیر از مهاجمان که دانش دقیق مقادیر دامنه و فرکانس رخدادشان را دارند پنهان می ماند.
- تقسیم و مقیاس بندی مقادیر رمز شده در پایگاه داده را توزیع پذیر یکنواخت می کند و از مهاجم برای تحلیل آماری جلوگیری می کند.
- چون ساختارهای مقداری و ساختاری سمت کارساز نگهداری می شوند بار پردازش پرس و جو سمت کارساز است.

معایب:

- مقیاس بندی سبب افزایش اندازه داده می شود که موجب زمان اضافه در پردازش پرس و جو می شود.
- نمی تواند امنیت را در برابر دانش پیشین توزیع برجسب، توزیع بارکاری پرس و جو و ارتباط بین مقادیر داده تهیه کند.
- روش در درج و به روز رسانی کارا نیست.

روش دوم

ایده اصلی این روش معرفی عبارات رمزنگاری قوی است. این عبارات به مشتری کمک می کنند یک رده غنی از سیاست های

یک قانون کلی وجود دارد که برای هر گره پدر p و گره بچه c داریم: کران راست $p < \text{کران راست } c$

کران چپ $p > \text{کران چپ } c$

برای نمایش ندادن ساختار XML شما ویرایش می شود و شاخص ساختاری ناپیوسته یا DSI^{۱۰} نامیده می شود. در DSI بازه [۰, ۱] به ریشه اختصاص می یابد. فرزندان زیربازه آن را می گیرند. بازه های فرزندان با یک الگوریتم در زمان اجرا تعیین می شود. قانون کلی که در بالا گفته شد حفظ می شود. DSI ساختار XML را از کارگزار مخفی می کند.

الگوریتم به دست آوردن DSI به صورت زیر است که w1 و w2 دو عدد حقیقی دلخواه و تصادفی هستند که برای هر گره انتخاب می شوند. Max و min بازه گره پدر است. [۳]

روش اول

در روش [۳] دو جدول برای شاخص ساختاری استفاده شده است. یکی جدول بلوک رمزنگاری و دیگری جدول DSI. جدول DSI برجسب ها را در یک ستون نگه می دارد و بازه های متناظر را در ستون دیگر، فقط برجسب های مجرمانه با استفاده از رمزنگاری OTP^{۱۱} رمز می شوند که باعث می شود روی گره هایی که رمز نشده اند پردازش کارای پرس و جو فراهم شود. درواقع بین امنیت و کارایی یک توازن برقرار شده است. از آنجا که ذخیره تمام بازه ها اطلاعات مهمی را در اختیار کارساز قرار می دهد گره های همسایه از یک برجسب که در یک بلوک رمزنگاری قرار دارند در هم ادغام می شوند. بدین ترتیب کارساز نمی تواند بفهمد که بازه نشان داده شده مربوط به یک گره است یا چند گره.

در این روش برای شاخص مقداری از درخت B استفاده شده است طوری که پرس و جوی بازه ای را پشتیبانی می کند. هر ورودی درخت B به شکل $\langle \text{evaluate}, \text{Bid} \rangle$ است که evaluate شکل رمز شده مقدار از پایگاه داده است و Bid، شناسه بلوک رمز شده است که شامل تعداد تکرارهای evaluate است.

این روش رمزنگاری سبب حملات مبتنی بر فرکانس می شود چرا که توزیع داده های رمز شده مانند رمز نشده است. به همین دلیل از تقسیم و مقیاس بندی استفاده می شود. به دلیل استفاده از تقسیم یک کلمه به کلمات متفاوتی رمز می شود و توزیع مسطحی از داده ها به دست می آید، مثلاً طبق ساختار uniform. تقسیم برای جلوگیری از حملات فرکانسی کافی نیست. با استفاده از مقیاس بندی اندازه دامنه هدف ضرب می شود. تعداد رخداد داده های رمز شده در یک فاکتور مقیاس ضرب می شود. هدف اصلی تقسیم و مقیاس بندی تغییر توزیع فرکانسی مقادیر داده رمز شده در شاخص مقداری است.

پردازش پرس و جو به این صورت است: ابتدا مبدل پرس و جو

10-Discontinuous Structural Index

11-One Time Pad

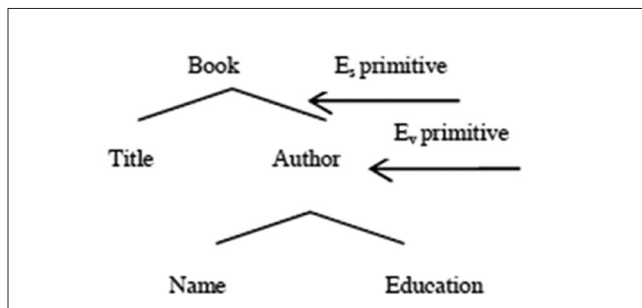
امنیتی برای داده XML مشخص کند. با این عبارات انتخاب می‌کند که داده حساس پنهان بماند. سه عبارت رمزنگاری پیشنهاد شده است: عبارت E_V (مقدار رمز) یک زیردرخت را رمز کرده و آن را با گره رمز شده جایگزین کرده است. مثلاً در شکل ۱۲ زیردرخت با ریشه «author» رمز شده و با یک گره رمز شده جایگزین می‌شود. E_T (برچسب رمز) فقط برچسب‌های زیردرخت با ریشه n (شامل گره n) را رمزنگاری می‌کند. E_S (ساختار رمز) رابطه بین دو گره مشخص شده را پنهان می‌کند. مثلاً E_S به رابطه بین Book و author اعمال می‌شود. این مدل شمای XML رمز نشده و سه عبارت رمزنگاری را به عنوان ورودی می‌گیرد و نمایش XML سمت کارساز را در خروجی می‌دهد. E_S ، E_V و E_T به ترتیب گفته شده اعمال می‌شوند. ایده دیگر این روش پیشنهاد یک راهبرد افزاز چند بعدی است. اطلاعاتی که در کارساز ذخیره می‌شود به عنوان یک فضای N بعدی دیده می‌شود. این فضای N بعدی به یک مجموعه از افزازها افزاز می‌شود. به هر افزاز یک id تصادفی داده می‌شود. افزازها کل دامنه را می‌پوشانند و همپوشانی ندارد. افزاز هم‌عرض برای جلوگیری از حملات مبتنی بر فرکانس استفاده می‌شود. راهبرد افزاز چند بعدی بر محدودیت‌های امنیتی تک بعدی غلبه می‌کند. [۲]

- اکثر پردازش پرس و جو سمت کارساز انجام می‌شود.
- امکان انجام پرس و جوهای بازه‌ای را می‌دهد.

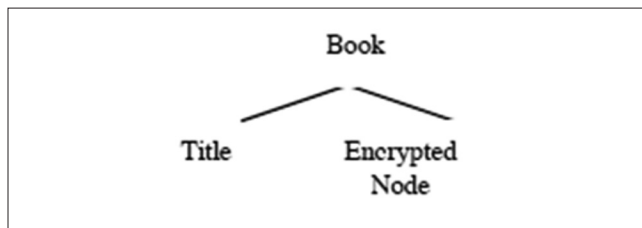
۲-۵. نگهداری شاخص‌ها سمت مشتری

در روش [۴] از روش XQEnc استفاده می‌شود. برای پردازش پرس و جو XML رمز شده استفاده می‌شود. XQEnc از فشردگی استخوانی^{۱۳} و برداری^{۱۲} استفاده می‌کند. در فشردگی برداری یک سند XML به بردارهای مسیر افزاز می‌شود که ترکیبی از برگ‌های غیر خالی هستند. در فشردگی استخوانی، افزونگی اسناد XML با استفاده از اشتراک زیرشاخه مشترک حذف می‌شود. شاخه‌های متوالی و یکسان با یک شاخه همراه با توضیح فراوانی جایگزین می‌شود. به این وسیله سند XML بسیار کوچکتر می‌شود. مثلاً سند XML شکل ۱۴ به شکل ۱۵ فشرده می‌شود.

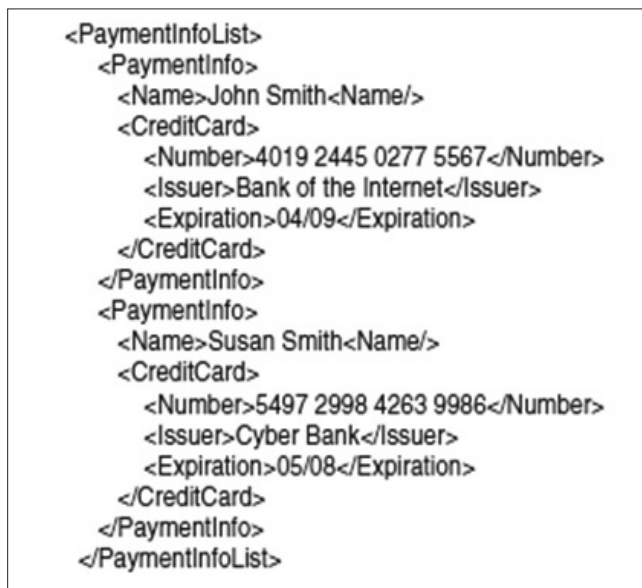
در روش XQEnc برای هر سند XML، یک استخوان فشرده شده S محاسبه شده و سمت مشتری ذخیره شده است. یک مجموعه از بردارهای داده متناظر D محاسبه شده و سمت کارساز ذخیره شده است. S هرگز با کارساز به اشتراک گذاشته نمی‌شود. پس ساختار سند XML از فرد ثالث مخفی شده است. برای هر شیء i در D یک سه گانه $\langle V_i, P_i, T_i \rangle$ ایجاد شده است. V_i بردار شامل P_i ، موقعیت سند و T_i مقدار متنی i را نمایش



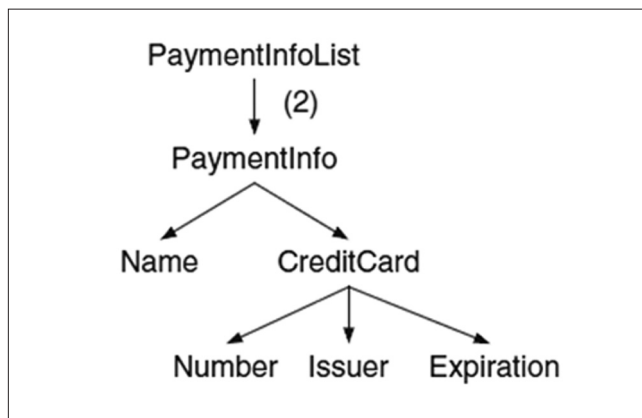
شکل ۱۲) شکل اولیه درخت



شکل ۱۳) شکل درخت پس از رمزنگاری



شکل ۱۴) یک سند XML



شکل ۱۵) فشردگی استخوانی سند شکل ۱۴

12-skeleton

13-vectorization

Path Schema ID	Path Schema
PS1	Bib/Book*/Author
PS2	Bib/Book*/Title
PS3	Bib/Book*/Author/Name
PS4	Bib/Book*/Author/Education
...	...

شکل ۱۷) جدول شناسه شمای مسیر

/PaymentInfoList/PaymentInfo/Name	[John Smith, Susan Smith]
/PaymentInfoList/PaymentInfo/CreditCard/Number	[4019 2445 0277 5567, 5497 2998 4263 9986]
/PaymentInfoList/PaymentInfo/CreditCard/Issuer	[Bank of the Internet, Cyber Bank]
/PaymentInfoList/PaymentInfo/CreditCard/Expiration	[04/09, 05/08]

شکل ۱۶) بردارهای مسیر سند شکل ۱۴

- با وجود استفاده از فشردگی استخوانی، احتمال مشکل مدیریت فضا وجود دارد مخصوصاً اگر مشتری حافظه محدودی دارد یا سند بزرگ و نامرتب باشد.

۳-۵. نگهداری شاخص‌ها سمت مشتری و کارساز

پردازش پرس و جوی رمز شده با نگهداری شاخص‌ها در هر دو سمت کارساز و مشتری مدیریت شده است. ما این رویه را به دلیل استفاده از یک روش جدید تحت یک عنوان متفاوت بررسی کرده‌ایم. [۲] شمای سند XML سمت مشتری ذخیره شده است. مسیرها با id منحصر به فردشان ذخیره می‌شوند که IDهای شمای مسیر نامیده می‌شود. $\times$ یعنی یک یا چند گره با برچسب هم‌نام می‌تواند وجود داشته باشد. با استفاده از $\times$ سند شما برای ذخیره در مشتری کوچک می‌شود.

سمت کارساز دو جدول درهم‌ساز وجود دارد. اولین جدول از نمونه‌های مسیر به عنوان کلید استفاده می‌کند (شکل ۲۰) و دومی از مقادیر مسیر (شکل ۲۱).

فرض کنید مشتری می‌خواهد پرس و جوی `/book [title='spring']/author/name` را انجام دهد. مشتری ابتدا شمای ذخیره شده در سمت خودش را جستجو می‌کند و می‌فهمد id شمای مسیر `PS2:bib/book*/title` است. مشتری تابع `getPathInstancesForValue` را محاسبه می‌کند سپس تابع `H(PS2-spring)` را با پارامتر `H(PS2-spring)` فراخوانی می‌کند. مقدار بازگشتی از کارساز `E({1}, k, 21)` است. مشتری این جواب را با استفاده از مقدار تصادفی همراه کلید رمزگشایی می‌کند و می‌فهمد جواب در اولین نمونه `{1}` کتاب در شمای XML است. سپس مشتری مسیر `title` را فیلتر می‌کند و مسیر `author/name` را به پرس و جو اضافه می‌کند. مشتری می‌داند که مسیر `PS3:author/name` است، پس پرس و جو `bib/book[1]/author/name` است که `PS3-1` است. مشتری تابع `GetValueforPathInstance` را با پارامتر `H(PS3-1)` فراخوانی می‌کند. در نهایت کارساز مقدار رمز شده را همراه با مقدار تصادفی برمی‌گرداند `E(F.WELL, k, 11)`. مشتری با کلید و مقدار تصادفی ۱۱ این جواب را رمزگشایی می‌کند و می‌فهمد `F.WELL` پاسخ است.

مزایا:

- ایده اصلی این روش استفاده از مقدار تصادفی بود که از حملات مبتنی بر فرکانس جلوگیری می‌کند چون متن‌های ساده یکسان به صورت متن‌های رمزبندی شده متفاوت رمز می‌شوند.

می‌دهد. سپس هر سه گانه به نمایش روبرو تبدیل می‌شود: $\langle \text{etuple}, V_i^e, P_i^e, T_i^e \rangle$ چندتایی رمز شده است و سایر ورودی‌ها شاخص‌های رمز شده متناظر سه گانه است. طبق XQEnc شاخص‌های رمزنگاری می‌تواند idهای باکت باشد یا مقادیر رمز شده با استفاده از رمزنگاری حفظ ترتیب باشد. بدین ترتیب بردارهای داده به یک جدول چهار ستونی تبدیل می‌شود.

الگوریتم XQEnc در سمت مشتری اجرا می‌شود. وقتی نیاز به دسترسی به یک شیء در بردارهای داده باشد، پرس و جو به کارگزار داده می‌شود:

```
SELECT etuple FROM R(V) WHERE Vs=cryptoindex(v) & Ps=cryptoindex(p)
```

اگر پرس و جو شامل شرط متنی یا بازه‌ای هم باشد این شرط در where اضافه می‌شود:

```
SELECT etuple FROM R(V) WHERE Vs=cryptoindex(v) & Ps=cryptoindex(p) & Ts=crptoindex ("string")
```

کارساز فقط به عنوان یک منبع خارجی فرض می‌شود. بعد از این که مشتری پرس و جو را فرستاد کارساز کارش را شروع می‌کند و نتایج را بازبایی و به مشتری می‌فرستد. مشتری آن را رمزگشایی کرده و برای پردازش بعد استفاده می‌کند.

مزایا:

- ایده اصلی این روش ذخیره شمای XML به عنوان یک استخوان فشرده شده در مشتری است که آن را برای کارساز غیر قابل دسترس می‌سازد. در آن حالت اطلاعات ساختاری از کارساز مخفی شده است.
- اگر در XQEnc از رمزنگاری حفظ ترتیب به جای باکت‌بندی استفاده شود پرس و جوهای بازه‌ای پشتیبانی می‌شوند.
- پرس و جوهای شامل عبارات معروف، این روش بسیار کاراست چون فقط داده لازم برای رمزگشایی را برای مشتری بازبایی می‌کند.

معایب:

- بار پردازشی سمت مشتری است که کارایی را کاهش می‌دهد.
- نیاز است که مشتری شاخص‌ها را نزد خود و در محیط توزیع شده نگه دارد.
- برای هر درج در پایگاه داده باید یک به روزرسانی در سمت مشتری انجام شود.

Cryptographic Hash (PS-V)	E(PI*, k, nonce)	Nonce
H(PS2-Spring)	E({1},k,21)	21
H(PS3-F.WELL)	E({1},k,22)	22
H(PS3-J.HAND)	E({2},k,23)	23
H(PS2-Football)	E({2},k,24)	24

شکل ۱۹) جدول نمونه‌های مسیر از روی مقادیر

Cryptographic Hash(PI)	E(value, k, nonce)	Nonce
H(PS2-1)	E(Spring,k,10)	10
H(PS3-1)	E(F.WELL,k,11)	11
H(PS3-2)	E(J.HAND,k,12)	12
H(PS2-2)	E(Football,k,13)	13
...	...	...

شکل ۱۸) جدول مقادیر از روی نمونه‌های مسیر

معایب:

به طور کلی هیچ روشی از تمامی انواع پرس و جوها پشتیبانی نمی‌کند.

اغلب روش‌های موجود روی بازبانی شاخص‌های ساختاری در پردازش پرس و جو تمرکز کرده‌اند، مدیریت شاخص‌ها در نظر گرفته نشده باید سازوکارهای کارایی برای بروزرسانی شاخص‌ها ارایه شود مخصوصاً در اسناد XML که زیاد به روز می‌شوند.

بهبود دیگر پشتیبانی از پرس و جوهایی^{۱۴} با قاعده است، مثلاً برای پاسخ به سوال $a[a-z]b$ نیاز به ۲۶ پرس و جو وجود دارد، یک سازوکار شاخص‌گذاری خوب در آینده ممکن است بتواند آن را انجام دهد.

برای بهبود کارایی پردازش پرس و جو در XML رمز شده از کارسازهای توزیع شده استفاده شود. [۲]

- نیاز به چند دور ارتباطی بین مشتری و کارساز دارد که پهنای باند را مصرف می‌کند و زمان پردازش پرس و جو را افزایش می‌دهد.
- امکان انجام پرس و جوهایی بازه‌ای وجود ندارد.
- مشتری باید پیوسته نتایج رمز شده را پردازش و توابع درهم‌ساز را محاسبه کند پس باید توانایی پردازشی قابل توجهی داشته باشد.

۶- نتیجه‌گیری و کارهای آینده

روش‌هایی که از ایجاد شاخص استفاده می‌کنند نیاز به هزینه ذخیره‌سازی برای ذخیره شاخص دارند. در روش باکت‌بندی که جواب اضافه زیاد است حجم ارتباطات بین مشتری و کارساز افزایش می‌یابد. در B^+ سربار ارتباطی بین مشتری و کارساز زیاد است. مسئله مهم دیگر در این روش‌ها بروزرسانی شاخص است که گاهی نیاز به شاخص‌بندی مجدد می‌باشد. پس این روش‌ها عمدتاً برای داده‌های فقط خواندنی مناسب می‌باشند.

رمزنگاری با حفظ ترتیب نسبت به حمله known plaintext امن نیست. جواب اضافه ندارد. محاسبات اولیه هزینه زیادی برای مشتری دارد و بروزرسانی هزینه زیادی دارد.

در زمینه رمزنگاری هم‌ریخت اختفایی فعالیت زیادی انجام نشده و پیدا کردن این توابع هم کار ساده‌ای نیست. [۷]

پشتیبانی از انواع پرس و جوها در این روش‌ها به صورت زیر است: [۵]

تجمعی	بازه‌ای	تساوی	روش
☒	☒	☒	Bucket-based
☒	☒	☒	Hash-based
☒	☒	☒	B+ Tree
☒	☒	☒	OPESS
☒	☒	☒	PH

از روش‌های گفته شده برای جستجو در XML هر کدام معایبی داشتند، مهمترین عیب روش اول هزینه بر بودن به روزرسانی آن به دلیل استفاده از تقسیم و مقیاس‌بندی بود. مهمترین مشکل روش دوم این بود که هزینه ذخیره و پردازش شاخص‌ها سمت مشتری انجام می‌شد و روش سوم هم فقط از پرس و جوهایی تساوی پشتیبانی می‌کرد.

مراجع

- [1] M. Gertz, S. Jajodia, «Handbook of Database Security-Applications and Trends», Springer Science+Business Media, 2008.
- [2] O. Unay, T.I. Gundem, «A Survey on Querying Encrypted XML Documents for Databases as a Service», SIGMOD RECORD, VOL 37, NUMB 1, pages 12-20, 2008.
- [3] H. Wang, L. Lakshmanan, «Efficient Secure Query Evaluation over Encrypted XML Databases», 32nd International Conference on Very Large Data Bases, September 12-15, 2006.
- [4] Y. Yang, W. Ng, H.L. Lau, J. Cheng, «An Efficient Approach to Support Querying Secure Outsourced XML Information», CAiSE, LNCS 4001, p:157-171, 2006.
- [5] S. Foresti, «Preserving Privacy in Data Outsourcing», Phd. Thesis, 2007.
- [6] E. Damiani, S. De Capitani di Vimercati, S. Jajodia, S. Paraboschi, P. Samarati, «Balancing Confidentiality and Efficiency in Untrusted Relational DBMSs», In Proceedings of the ACM conference on computer and communications security, CCS 2003, Washington, October 27-31, 2003.
- [7] R. Agrawal, J. Kiernan, R. Srikant, Y. Xu, «Order preserving encryption for numeric data», In SIGMOD, 2004.
- [۸] م.ع. هادوی، «پرس و جو روی داده‌های رمز شده در مدل پایگاه داده به عنوان خدمت»، دانشگاه صنعتی شریف، ۱۳۸۸.

14-Regular expression

به یاد معلمی که شاگردانش او را استاد می نامند

سید ابراهیم ابطاحی

عضو هیأت علمی دانشکده مهندسی کامپیوتر - دانشگاه صنعتی شریف

پست الکترونیکی: abtahi@sharif.edu

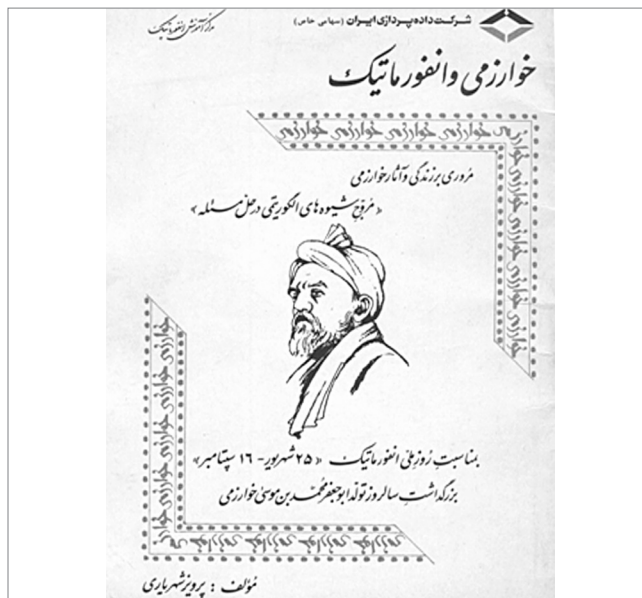
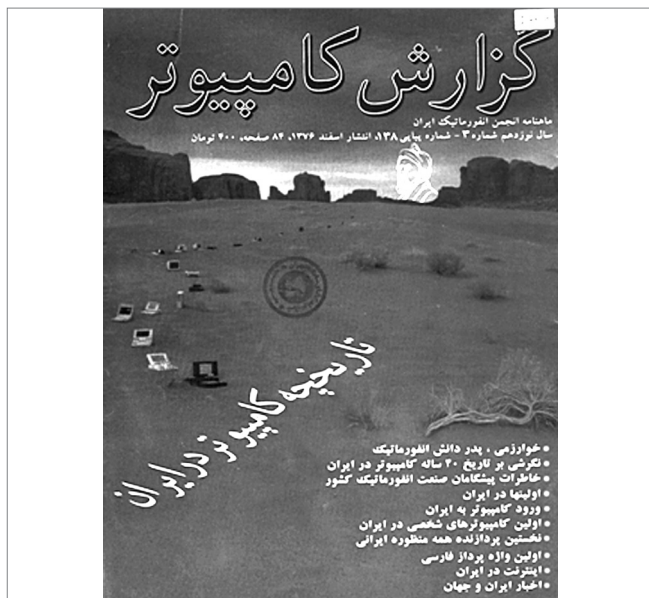
- استاد «پرویز شهریاری» بامداد جمعه، ۲۲ اردیبهشت در بیمارستان جم تهران بر اثر سکته قلبی چشم از جهان فانی فروبست و پیکرش در آرامگاه زرتشتیان تهران در قصر فیروزه آرام گرفت. جستجوی ۲۲
- صدم ثانیه‌ای در گوگل ۵۲۲ هزار سند در رابطه با نام و یاد جاویدان او در اختیار همگان می‌گذارد، شامل زندگینامه مفصل و نوشته‌هایی از او، به یاد و برای او. با ذکر چند نقل قول از او به ذکر خاطره‌ای درباره او می‌پردازم که مرتبط با رایانه و فا و ثبت‌کننده نام استاد در معرفی پژوهش محور یکی از پیشگامان جهانی این رشته یعنی ابوجعفر محمد بن موسی خوارزمی است.
- توصیه خردمندانه استاد به مؤلفین و نویسندگان که: «بی سر و صدا کار کنید» را با ذکر اسامی برخی از تألیفات و ترجمه‌های استاد پی می‌گیریم تا نتایج تلاش فوق انسانی این نحل از اساتید نسل گذشته برای تأثیرگذاری فرهنگی را نشان دهیم:
- دانش‌نامه ریاضی ایران
- تاریخ حساب، رنه تاتون، انتشارات امیرکبیر، چاپ اول ۱۳۲۹، ترجمه.
- ریاضیات در شرق، انتشارات خوارزمی، ۱۳۵۲، ترجمه.
- سرگذشت آنالیز ریاضی، انتشارات امیرکبیر، ۱۳۵۴، ترجمه.
- ریاضیات کار بسته، انتشارات هدهد، ۱۳۶۰، ترجمه.
- لباچفسکی و هندسه ناقلیدسی، انتشارات توکا، ۱۳۶۰، تالیف.
- پویایی ریاضیات، انتشارات پویش، ۱۳۶۰، ترجمه.
- اواریسست گالوا، چاپ اول ۱۳۶۴ انتشارات هدهد، چاپ دوم ۱۳۷۳ نشر بردار، ترجمه.
- من ریاضیدانم، نوربرت وینر، انتشارات فاطمی، چاپ اول ۱۳۶۴، ترجمه.
- آفرینندگان ریاضیات عالی، ل. س. فریمان، انتشارات فردوسی، چاپ اول ۱۳۶۳، ترجمه.
- خوارزمی و انفورماتیک، شرکت داده‌پردازی ایران، ۱۳۷۰، تالیف.
- خلاقیت ریاضی، جورج پولیا، انتشارات فاطمی، ۱۳۷۳، چاپ چهارم، ترجمه.
- عالی جناب چکمه (گوشه‌ای از تاریخ ریاضیات)، انتشارات پژوهنده، چاپ اول ۱۳۷۸، چاپ دوم ۱۳۸۴، تالیف.



- مثلثات مستقیم الخط و کروی، س. ای. نووسلو، چاپ چهارم ۱۳۷۸
- نشر دانش امروز، ترجمه.
- روش‌های جبر، دو جلد، چاپ اول، انتشارات امیر کبیر، ۱۳۴۴،
- تالیف.
- اعداد اول، امیل بورل، انتشارات امیر کبیر، چاپ اول ۱۳۴۴، چاپ
- سوم ۱۳۸۱، ترجمه.
- جبر از آغاز تا پایان (خودآموز)، واویلف / ملنیکف / آلکس نیک /
- پاسی چنکو، انتشارات تهران، چاپ اول ۱۳۶۹، چاپ دوم زمستان
- ۱۳۷۱، ترجمه..
- تقارن در جبر، و. گ. بالتیانسکی / ن. یا. ویلنکین، انتشارات
- امیرکبیر، چاپ مرداد ۱۳۴۷، ترجمه.
- هندسه غیراقلیدسی، نشر اندیشه، چاپ سوم اردیبهشت ۱۳۵۲،
- ترجمه.
- ورودی به نظریه مجموعه‌ها، ژ. بروئر، انتشارات پویش، چاپ اول
- ۱۳۵۹، ترجمه.
- استقرار ریاضی، نوشته سومینسکی و گولووینا و یاگوم، انتشارات
- خوارزمی، چاپ اول خرداد ماه ۱۳۴۸، چاپ دوم آذر ماه ۱۳۶۵،
- چاپ سوم آبان ماه ۱۳۷۷.
- نظریه مجموعه‌ها، واتسلو سرپینسکی، انتشارات خوارزمی، چاپ
- اول ۱۳۵۰، چاپ سوم مهرماه ۱۳۶۴، ترجمه.
- نامساوی‌ها، پاول پتروویچ کار و کین، انتشارات خوارزمی، ۱۳۵۰،
- ترجمه.
- اشتباه استدلال‌های هندسی، انتشارات خوارزمی، ۱۳۵۰، ترجمه.
- ورودی به منطق ریاضی، انتشارات خوارزمی، ۱۳۵۴، ترجمه.
- انعکاس، انتشارات خوارزمی، ۱۳۵۱، ترجمه.

- سرگذشت ریاضیات، نشر مهاجر، چاپ اول ۱۳۷۸، تالیف.
- لگاریتم (تاریخ استدلالی لگاریتم)، گ. ک. استاپو، انتشارات
- خوارزمی، چاپ اول ۱۳۴۸.
- هندسه در گذشته و حال، انتشارات امیرکبیر، سال‌های پنجاه،
- ترجمه.
- غیاث‌الدین جمشید کاشانی ریاضیدان ایرانی، انتشارات فنی ایران،
- ۱۳۷۸، تالیف.
- جوهر، روش و کارآیی ریاضیات، ۳ جلد، انتشارات فنی ایران،
- ۱۳۸۰، ترجمه.
- مسئله‌های تاریخی ریاضیات، و. د. چیسستیاکوف، نشر نی،
- ترجمه.
- فلسفه، اخلاق و ریاضیات، انتشارات پژوهنده، چاپ نخست ۱۳۸۰،
- ترجمه و تالیف.
- خلاقیت در ریاضیات و مهندسی، انتشارات پژوهنده، چاپ اول
- ۱۳۸۰، تألیف و ترجمه.
- ریاضیات و هنر، انتشارات پژوهنده، چاپ نخست ۱۳۸۱، ترجمه و تألیف.
- آموزش ریاضی، نشر مهاجر، چاپ اول ۱۳۸۴، ترجمه و تألیف.
- گاهنامه ریاضی، شامل شرح حال و نظر ریاضی دانان، انتشارات
- مهاجر، ۱۳۸۰، تالیف.
- شما هم می‌توانید در درس ریاضی خود موفق باشید، انتشارات
- مدرسه، چاپ اول ۱۳۷۸، چاپ دوم ۱۳۸۰، تألیف.
- نگاهی به تاریخ ریاضیات در ایران، شرکت انتشارات علمی و
- فرهنگی، چاپ نخست بهار ۱۳۸۵، تألیف.
- روش مختصات، ل. س. پونتریگین، نشر پژواک کیوان، چاپ اول
- پاییز ۱۳۸۲، ترجمه.

- نظریهٔ ساختمان‌های هندسی، آگوست آدلر، انتشارات فردوس، چاپ اول ۱۳۶۹، ترجمه.
- هندسه پرگار، انتشارات دانشجو، سال‌های شصت، ترجمه.
- عبارات‌های متقارن در جبر مقدماتی، رز نشر، سال‌های شصت، تالیف.
- قدر مطلق در حوزه عددهای حقیقی، رز نشر، سال‌های شصت، تالیف.
- آنالیز برداری و نظریه میدان، انتشارات فاطمی، سال‌های شصت، ترجمه.
- قضیهٔ مستقیم و قضیهٔ معکوس، ا. س. گراوشتین، نشر نی، چاپ اول ۱۳۷۵، ترجمه.
- تابع‌های متناوب، رز نشر ۱۳۶۸، تالیف.
- بخش درست عدد $[x]$ ، رز نشر ۱۳۶۸، نشر مهاجر ۱۳۷۸، تالیف.
- روش استقرای ریاضی، رز نشر، ۱۳۶۸، تالیف.
- ورودی به نظریه آنالیز ترکیبی، رز نشر، ۱۳۶۸، تالیف.
- بسط دو جمله‌ای با نمای طبیعی، رز نشر، ۱۳۶۸، تالیف.
- تربیع دایره و غیرجبری بودن عدد پی، رز نشر، ۱۳۶۸، تالیف.
- ورودی به نظریه احتمال، رز نشر، ۱۳۶۸، تالیف.
- آنالیز ریاضی، رز نشر، ۱۳۶۸، ترجمه.
- لگاریتم، گ. استاکوف، انتشارات خوارزمی، سال‌های پنجاه، ترجمه.
- آنالیز ریاضی، ۳ جلد (با همکاری باقر امامی)، انتشارات فردوس، ۱۳۶۸، ترجمه.
- قضیهٔ فرما، م. م. پوستنیکوف، نشر نی، چاپ اول ۱۳۷۹، ترجمه.
- بنیان‌های هندسه، و. ای. کوستین، نشر مهاجر، چاپ اول ۱۳۸۱، ترجمه.
- آنالیز برداری، آ. آ. گولدفاین، انتشارات فاطمی، چاپ اول ۱۳۶۴، چاپ دوم ۱۳۶۸، ترجمه.
- بازی با بی‌نهایت، روزا پتر، چاپ اول ۱۳۵۶ انتشارات توکا، چاپ دوم ۱۳۶۳ انتشارات فردوسی، ترجمه.
- منحنی‌ها در فضا، دونووان جونسون، انتشارات چاپار، چاپ اول ۲۵۳۶، ترجمه.
- دستگاه‌های محدود ریاضیات، انتشارات چاپار، ۲۵۳۴، ترجمه.
- داستان مجموعه‌ها، ن. ی. ویلنکین، انتشارات توکا، چاپ اول ۱۳۵۵، چاپ دوم شهریور ماه ۱۳۵۷، ترجمه.
- داستان‌های ریاضی، انتشارات توکا، سال‌های شصت، ترجمه.
- روش مختصاتی و هندسهٔ چهار بعدی، ی. م. گلفاند/ا. گ. گلاگوله/ا. آ. آ. کیریلوف، انتشارات خوارزمی، ۱۳۵۶، ترجمه.
- مسیر ریاضیات جدید، و. و. سایر، رز نشر/سازمان چاپ و نشر مشهد، چاپ دوم بهار ۱۳۶۹، ترجمه.
- مهم‌ترین مسأله‌ها و قضیه‌های ریاضی، شکلیارسکی/چنتسوف/یاگلووم، انتشارات مجید/انتشارات فردوس، چاپ دوم ۱۳۷۴، ترجمهٔ پرویز شهرباری و ابراهیم عادل.
- مسأله‌های دشوار ریاضی، کنسانتین شاخنو، انتشارات فردوس، چاپ سوم ۱۳۷۴، ترجمه.
- مسابقه‌ها، کنکورها و المپیادهای ریاضی، انتشارات جاودان خرد، چاپ اول ۱۳۷۲، تالیف.
- گزیدهٔ مسأله‌های تازه و بکر مقدماتی ریاضیات، و. پلاتونف/ک. ر. لیوک/و. زارتسکی/ن. مدتلسکی/ل. توتایف، نشر پژواک کیوان، چاپ اول ۱۳۸۲، ترجمه.
- مسائل مسابقات ریاضی، و.ا. س. کوشچنکو، انتشارات امیر کبیر، چاپ هشتم ۱۳۶۵، ترجمه.
- روش‌های مثلثات (با همکاری آقای فیروزنیا)، انتشارات خوارزمی، سال‌های پنجاه، تالیف.
- تمرین‌ها و مسائل آنالیز ریاضی، ب. ب. دمیدوویچ، انتشارات امیر کبیر، چاپ ششم ۱۳۸۲، ترجمه.
- مسأله‌های ریاضی، آسان ولی...، انتشارات توکا، ۱۳۵۴، ترجمه.
- مسأله‌های المپیادهای مجارستان، انتشارات دانشجو، سال‌های شصت، ترجمه.
- دربارهٔ حد، آ. آ. کیریلوف انتشارات آزاده، ۱۳۶۳، ترجمه.
- تئوری اعداد (۲۵۰ مسألهٔ حساب)، نوشتهٔ واتسلاو سرپینسکی، انتشارات خوارزمی، چاپ اول آبان ماه ۱۳۴۹، چاپ دوم شهریور ماه ۱۳۶۹، چاپ سوم تیر ماه ۱۳۷۷.
- مسأله‌های المپیادهای آمریکا، (با همکاری آقای عادل)، نشر بردار، ۱۳۶۸، ترجمه.
- المپیاد ریاضی لنینگراد (از سال ۱۹۶۱ به بعد)، د. و. فومین، چاپ اول انتشارات اینشتین ۱۳۷۴، چاپ دوم نشر گستره ۱۳۷۹، ترجمه.
- المپیادهای بین‌المللی (با همکاری آقای عادل)، انتشارات فاطمی، ۱۳۶۸، ترجمه و تالیف.
- مسأله‌های المپیادهای ریاضی در کشورهای مختلف، انتشارات فردوس، ۱۳۶۸، ترجمه.
- آمادگی برای المپیادهای ریاضی، انتشارات فاطمی، ۱۳۶۹، ترجمه.
- مسأله با حل (با همکاری آقایان امامی. حریرچی)، سال‌های چهل، تالیف و ترجمه.
- دوره اختصاصی جبر مقدماتی، سال‌های چهل، انتشارات امیرکبیر، ترجمه.
- مسائل امتحانی جبر چهارم دبیرستان‌های کشور با حل، انتشارات امیر کبیر، ۱۳۴۴، تالیف.
- مسائل امتحانی جبر پنجم دبیرستان‌های کشور با حل، انتشارات امیر کبیر، ۱۳۴۴، تالیف.
- تست حساب استدلالی (با همکاری آقایان امامی و قوام‌زاده)، انتشارات امیر کبیر، ۱۳۴۳، تالیف.
- مسائل جبر و راهنمای حل آن‌ها برای کلاس‌های کنکور (با همکاری آقای امامی)، انتشارات امیر کبیر، سال‌های چهل، تالیف.
- مسائل مثلثات و راهنمای حل آن‌ها برای داوطلبان کنکور (با



- کتابی در باره کتاب، سرگی لهوو، ترجمه.
- داستان‌های علمی، مارک تواین / ایزاک آسیموف... انتشارات فردوسی، چاپ اول خرداد ۱۳۶۰، ترجمه.
- علم، جامعه و انسان، جلد یک و دو، انتشارات هدهد، چاپ اول جلد دوم خرداد ۱۳۶۰، ترجمه و تألیف.
- آواربست گالوا، لئوپولد اینفلد، انتشارات هدهد، چاپ اول ۱۳۶۰، ترجمه.
- یک روز زندگی پسرک قبطی، ماتیو، انتشارات توکا، ترجمه.
- اخلاق و انسان، الگنا تانونا کروتووا، انتشارات فردوسی، چاپ دوم ۱۳۶۱، ترجمه.
- نظریه نسبیت در مسئله‌ها و تمرین‌ها، الکسی نیکلایه ویچ مالینین، نشر نی، چاپ دوم ۱۳۷۴، ترجمه.
- در جستجوی هماهنگی، الگ موروز، نشر مهاجر، چاپ اول ۱۳۸۲، ترجمه.
- در این مجموعه نام کتاب خوارزمی و انفورماتیک درج شده است که ادامه این نوشته بیان نحوه شکل‌گیری و نشر این نوشته است که در قالب یک خاطره، به بهانه آن مروری بر خصلت‌های معلمینی می‌پردازیم که همواره بوده و هستند و چراغ علم آغشته به فرهنگ و انسان دوستی را در این ملک روشن نگه داشته و نگه می‌دارند.
- به یاد می‌آورم که مرداد ماه ۱۳۷۱ بود و من که مرکز آموزش انفورماتیک را به تازگی در شرکت داده‌پردازی ایران برای ادامه پژوهش‌های موردی در زمینه آموزش انفورماتیک راه‌اندازی و اداره می‌کردم از طریق مدیریت وقت این شرکت (که اگر اشتباه نکنم آقای مهندس مسیح قائمیان بود) به دفتر ایشان دعوت شدم و مسئله برگزاری روز ملی انفورماتیک برای اولین بار در ایران با همکاری مرکز ما پیشنهاد شد. مسئله این بود بنا به
- همکاری آقای امامی، انتشارات امیر کبیر، سال‌های چهل، تألیف.
- مسائل هندسه و راهنمای حل آن‌ها برای داوطلبان کنکور (با همکاری آقای ازگمی)، انتشارات امیر کبیر، سال‌های چهل، تألیف.
- تست‌های ریاضیات (با همکاری آقای تقوی)، انتشارات امیر کبیر، سال‌های پنجاه، تألیف.
- تئوری اعداد (با همکاری آقای قوام‌زاده)، انتشارات امیر کبیر، سال‌های پنجاه، تألیف.
- حل مسائل آنالیز (با همکاری آقایان امامی و عصار)، انتشارات دانشگاه تهران، سال‌های پنجاه، تألیف.
- تست ریاضیات (با همکاری آقایان امامی و قوام‌زاده)، انتشارات امیر کبیر، ۱۳۵۰، ترجمه.
- مسئله‌های ریاضیات عمومی با حل (با همکاری آقای امامی)، انتشارات امیر کبیر، ۱۳۵۳، تألیف.
- مسئله‌های کنکور شوروی، انتشارات پویش، ۱۳۶۱، ترجمه.
- مسائل مسابقات ریاضی شوروی، انتشارات امیر کبیر، ترجمه.
- سرگرمی‌های هندسه، ی. ای. پرلمان، انتشارات خوارزمی، ترجمه.
- سرگرمی‌های جبر، ی. ای. پرلمان، انتشارات امیر کبیر، ترجمه.
- سرگرمی‌های ریاضی، ی. ای. پرلمان، ترجمه.
- ۱۷۵ مسئله منطقی، دی یو دبیزام / یانوش هرتسگ، نشر نی، چاپ اول ۱۳۶۶، چاپ دوم ۱۳۷۴، ترجمه.
- سرگرمی‌های توپولوژی (توپولوژی عمومی)، استفن بار، نشر نی، چاپ دوم ۱۳۷۴، ترجمه.
- در پی فیثاغورث، ش. النسکی، انتشارات امیر کبیر، چاپ پنجم ۱۳۸۴، ترجمه.
- در قلمرو ریاضیات، آ. پ. دوموریاد، انتشارات امیر کبیر، چاپ اول ۱۳۴۸، چاپ دوم ۱۳۶۳، ترجمه.
- باد و باران زاهاریا استانکو (رمان دو جلدی)، ترجمه.

جانکاه دیوان سالاری هنوز نشر نشده‌اند و این امر بیانگر این نکته است که کتاب‌های چاپ شده ایشان تنها می‌تواند بخش کوچکی از میراث فرهنگی این استاد باشد بنابراین آرزو می‌کنیم فرزندان نسبی و سببی استاد برای چاپ این گنجینه‌های فرهنگی نشر نشده اقدام کنند.

خلاصه پژوهش استاد درباره خوارزمی در شهریور ماه ۱۳۷۱ در سی صفحه با عنوان "زمانه: زندگی و آثار خوارزمی در مجموعه‌ای با نام "خوارزمی و انفورماتیک" با عناوین فرعی "مروری بر زندگی و آثار خوارزمی - مروج شیوه‌های الگوریتمی در حل مسئله - به مناسبت روز ملی انفورماتیک - ۲۵ شهریور - ۱۶ سپتامبر - بزرگداشت سالروز تولد ابوجعفر محمد موسی خوارزمی مؤلف: پرویز شهریاری" در پانصد نسخه منتشر شد. افتخار نوشتن مقدمه این مجموعه با عنوان "خوارزمی و انفورماتیک" را من داشتم ولی ناشرین این مجموعه ذیل صفحه اول آن نوشتند: چاپ مطالب این مجموعه ضرورتاً به معنای تأیید مواضع و اندیشه‌های نویسندگان آن نیست.

استاد در این نوشته بخش اول را به بررسی دوران پیش از خوارزمی اختصاص داده بود و در آن ابتدا به بررسی پیشینه ریاضیات در دوران باستان و پس از آن در چین و هند، در عصر دیوفانت، در اروپای غربی در ایران و در انتها در خوارزم پرداخته بود. بخش دوم به خوارزمی و کارهای او اختصاص داشت. این نوشته بسیار عمیق و پژوهش‌محور بود که خوشبختانه در شماره ۱۸۳ گزارش کامپیوتر که ویژه‌نامه تاریخی کامپیوتر در ایران بود تجدید چاپ شد.

نکته عبرت‌آموز آن که پیشنهاد دعوت از استاد برای ارائه پژوهش خود در باره زندگی خوارزمی در جلسه مزبور مورد موافقت واقع نشد و متأسفانه حتی به شکل رسمی برای حضور در آن جلسه از ایشان دعوت به عمل نیامد و تلاش جانکاهی صرف شد تا ایشان برای دریافت حق‌الزحمه ناچیز این پژوهش که هیچ‌گاه راجع به آن صحبت نکردند و خواهان دریافت آن نبودند ناچار به حضور در واحد مالی شرکت داده‌پردازی نشوند. در جلسه‌ای که برای تشکر از زحمات ایشان، تقدیم نسخه‌ای از نوشته منتشر شده و تقدیم حق‌الزحمه ناچیز پژوهش خدمتشان رسیدم و متأسفانه سومین و آخرین ملاقات ما بود، ایشان گفتند برای احترام به زحمات ما در برگزاری این مراسم فرهنگی به عنوان شنونده در جلسه فوق حضور داشته‌اند که این خود درس دیگری بود. این برخورد برای عبرت‌آموزی قابل مقایسه با جلسات متعددی برای قدرشناسی از استاد در سال‌های بعد است که در شرح حال ایشان قابل مطالعه است و دریغ از ما که قدر این بزرگواران را، به موقع در حیات مادیشان نمی‌دانیم و فرصت‌های بی‌نهایتی را بدین ترتیب از جامعه خود می‌گیریم، روانش شد.

پیشنهاد ایران و پذیرش یونسکو روز ۱۶ سپتامبر مقارن با ۲۵ شهریور به عنوان روز تولد خوارزمی از سوی یونسکو پذیرفته و اعلام گردیده بود و توصیه گردیده بود اعضای یونسکو هر سال این روز را به عنوان بزرگداشت نام و مقام این دانشمند بزرگ که اولین بار دستورالعمل‌های حل مسائل ریاضی را در گام‌های متوالی بیان و منظم کرده بود و بعدها واژه الگوریتم از نام الخوارزمی او به همین علت استخراج شده بود را جشن بگیرند. ایران در این سال به همت شورای عالی انفورماتیک کشور که اگر اشتباه نکنم قائم مقامی آن را آقای مهندس برات قنبری به عهده داشت در صدد برگزاری این مراسم بود. با توافقات انجام شده در صدد برگزاری این جشن بر آمدیم. ابتدا به گردآوری آنچه در مورد سرگذشت این دانشمند بزرگ موجود بود پرداختم و نوشته‌هایی از اساتید احمد آرام، ابوالقاسم قربانی، حسین خدیو جم، محمد محیط طباطبائی، محمد تقی دانش‌پژوه، علیرضا و احمد جعفری نائینی را فراهم آوردم. اما این مطالب برای یک ویژه‌نامه ناکافی بود و در ضمن چون همگی قبلاً به کرات چاپ شده بودند، تکراری و غیراصیل می‌نمود. در بین این نوشته‌های ناشانی از انجام تحقیقات توسط استاد شهریار یافتم که پیش از آن غیرشناخت به عنوان خواننده آثار برخوردی با استاد نداشتم. در پی ملاقات با استاد برآمدم از طریق استاد پرکار و دانای محل تحصیل دانشگاهی در دوره کارشناسی - مدرسه عالی برنامه‌ریزی و کاربرد کامپیوتر - جناب آقای پرویز ملک‌پور نماینده اسبق زرتشتیان در مجلس شورای اسلامی و صاحب امتیاز ماهنامه وزین چیستا که از ابتدا سردبیر آن بود افتخار ملاقات با استاد را در دفتر چیستا یافتم. سلوک استاد در همین برخورد کوتاه با منی که هیچ آشنائی قبلی با ایشان نداشتم آموختنی بود. استاد پس از شنیدن درخواست من برای در اختیار گذاشتن یا نوشتن زندگینامه‌ای از خوارزمی برای ارائه در این جشن، متواضعانه ابراز کردند که بیش از سه هزار صفحه دست نوشته تحقیقاتی در مورد این دانشمند دارند که هیچگاه فرصت نشر آن را نیافته‌اند و در مقابل درخواست زیادخواهانه من برای در اختیار گذاشتن خلاصه‌ای چند ده صفحه‌ای از آن که در این فرصت کم بتوان آن را منتشر کرد، گشاده‌رو قبول زحمت کردند و فرصتی ده روزه برای فراهم کردن آن خواستند. پس از این مدت دست نوشته استاد با خط زیبا و دقت پژوهش‌گرانه را - علیرغم بیماری و مشکلاتی که مطلع بودم آن هنگام با آن درگیر بودند - در زمانی که گفته بودند از ایشان گرفتم و همان شب بارها خواندم و از عمق گستردگی دانش و زبان نگارشی و دیگر ویژگی‌های این نوشته درس‌ها گرفتم که امیدوارم روزی این دست‌نوشته منتشر شود. سال‌ها بعد منتظر بودم متن کامل این پژوهش استاد منتشر شود اما این نوشته نظیر بسیاری از نوشته‌های فرهیختگان و اساتید ما پس از سال‌ها در چرخه

ارائه یک پروتکل امنیت مسیریابی در شبکه‌های موردی

علی یزدانیان

پشتیبان شبکه، شرکت تفسیرگران سیستم

پست الکترونیکی: ali_yazdanian_800@yahoo.com

سیامک لونی

دانشجوی کارشناسی ارشد، دانشگاه سراسری اراک

پست الکترونیکی: yooka_806@yahoo.com

۱- مقدمه

شبکه‌های موردی به دلیل توانایی در ارائه کاربردهای تجاری و چندرسانه‌ای از مقبولیت خاصی برخوردارند. شبکه‌های اقتضایی نمونه نوینی از شبکه‌های مخابراتی بیسیم هستند که به خاطر مشخصات منحصر به فردشان امروزه بسیار مورد توجه قرار گرفته‌اند. در این شبکه‌ها هیچ پایگاه مبنای تقویت کننده و مرکز سوئیچینگ ثابتی وجود ندارد، بلکه این خود گره‌ها هستند که عملیات تقویت داده سوئیچینگ و مسیریابی را انجام می‌دهند. با توجه به تغییرات مداوم در همبندی شبکه به واسطه تحرک گره‌ها، پروتکل‌های مسیریابی باید با اتخاذ نوعی راهبرد سازگار، این تغییرات را پشتیبانی کنند به نحوی که داده ارسالی به سلامت از مبدأ به مقصد برسد. از اوایل دهه ۸۰ میلادی تاکنون پروتکل‌های مسیریابی فراوانی برای

چکیده

ارائه سرویس‌هایی که همه جا و همه وقت در دسترس باشند، از چالش‌های مهم شبکه‌های کامپیوتری است. شبکه‌های موردی^۱ با توجه به مشخصات منحصر به فردشان گزینه‌ای مناسب جهت استفاده در میدان جنگ، مناطق دورافتاده، موقعیت‌های اورژانسی، کلاس‌های مجازی و ... هستند. با توجه به گسترش روزافزون این شبکه‌ها در عرصه‌های مختلف، لزوم طراحی بهینه پروتکل‌های مسیریابی بیش از پیش حائز اهمیت است. تاکنون پروتکل‌های مسیریابی زیادی برای شبکه‌های اقتضایی پیشنهاد شده است. هدف از طراحی این پروتکل‌ها کمینه‌سازی یک یا چند عامل محدودکننده راندمان شبکه (مثل محدودیت توان عملیاتی گره‌ها، سربارهای کنترلی، سربارهای پردازشی و ...) است. در این مقاله روی یکی از پروتکل‌های مسیریابی و امنیت آن متمرکز می‌شویم.

کلید واژه: پروتکل‌های مسیریابی، شبکه‌های موردی، SAODV.

1- Ad Hoc Network

۳- فواید مسیریابی چند مسیری

مسیرهای چندگانه می‌توانند تعادل بار، تحمل پذیری خطا، برآورده کردن نیازهای کیفیت، افزایش قابلیت اطمینان انتقال داده‌ها و پهنای باند بالا را فراهم کنند. تعادل بار می‌تواند با پخش ترافیک در طول چندین مسیر به دست آورده شود. استفاده از مسیریابی چندگانه در شبکه‌های سیار موردی با تراکم بالا، گذردهی بهتری را نسبت به مسیریابی تک مسیری دارد. پروتکل‌های مسیریابی چندمسیری بر روی تأخیر انتها به انتها و نرخ تحویل بار اثر می‌گذارند. از آنجایی که برآورده کردن این نیازها توسط روش‌های چندمسیری صورت می‌گیرد، در نتیجه جزء دسته الگوریتم‌های تنظیم بار محسوب شوند [۲].

۴- نحوه مسیریابی در شبکه‌های سیار موردی

برای ایجاد یک مسیر بین مبدأ و مقصد، یک جریان محدود شده عبور می‌کند که عموماً درخواست مسیر (RREQ) نامیده می‌شود. مبدأ ممکن است درخواست‌های مسیر چندتایی را دریافت کند. با این حال یک مسیر بر اساس کوتاه‌ترین فاصله انتخاب و پاسخ مسیر (RREP) را به مبدأ باز می‌فرستد. مسیرهای یافت شده در طول مرحله ایجاد مسیر (RREQ/RREP) معمولاً کوتاه‌ترین مسیرها هستند. به عنوان مثال، بهترین مسیر، مسیری است که کمترین تعداد جهش را داشته باشد. این مسیر به عنوان مسیر اصلی برای انتقال داده انتخاب می‌شود. در حالی که دیگر مسیرها فقط زمانی مورد استفاده قرار می‌گیرند که مسیر اولیه خراب شده باشد.

۵- نیازمندی امنیتی شبکه‌های موردی

یک الگوریتم مسیریابی خوب در شبکه موردی می‌بایست یک مسیر را به درستی بنا کند و از آن نگهداری نماید. بدین معنا که اجازه ندهد تا گره‌های متخاصم از ساخت یا نگهداری صحیح مسیر جلوگیری نمایند. در مجموع اگر یک الگوریتم نکات زیر را رعایت نماید می‌توان آن را یک الگوریتم امن نامید:

- × سیگنال‌های مسیریابی نتوانند جعل شوند.
- × سیگنال‌های دستکاری شده نتوانند به داخل شبکه تزریق شوند.
- × پیام‌های مسیریابی در طی انتقال بجز در روند عادی پروتکل تغییر پیدا نکنند.
- × حلقه‌های مسیریابی در طی فعالیت‌های متخاصم ایجاد نشوند.
- × کوتاه‌ترین مسیرها توسط گره‌های متخاصم تغییر پیدا نکنند.
- × گره‌های غیرمجاز می‌بایست از شبکه کنار گذاشته شوند.
- × همبندی شبکه نباید توسط مدیر، به هیچ کدام از گره‌های مجاز و یا متخاصم نشان داده شود [۳].

شبکه‌های اقتضایی پیشنهاد شده است. این پروتکل‌ها طیف وسیعی از مبانی و روش‌های طراحی را شامل می‌شوند. از یک تعریف ساده برای پروتکل‌های اینترنتی گرفته تا روش‌های سلسله مراتبی چند سطحی پیچیده. بسیاری از این پروتکل‌ها بر اساس فرض‌های ابتدایی ساده‌ای طراحی می‌شوند. مثلاً در بسیاری از این پروتکل‌ها، تمام گره‌ها منابع و توانایی‌های یکسانی دارند. به عنوان نمونه، برخی از گره‌ها طیف ارسالی یکسانی دارند و پیوندها دوطرفه فرض می‌شوند. البته پروتکل‌هایی هم هستند که عملاً با پیوندهای یک طرفه سروکار دارند. نهایتاً اگر چه هدف بسیاری از پروتکل‌ها قابل استفاده بودن برای شبکه‌های بزرگ است، اما معمولاً و به طور میانگین برای ۱۰ تا ۱۰۰ گره طراحی می‌شوند. کمبود توان باتری، پهنای باند محدود، میزان خطای زیاد، تراکم داده‌ها و تغییرات مداوم در همبندی شبکه از مشکلات اساسی شبکه‌های اقتضایی است. کیفیت سرویس و امنیت، به دلیل نبود روش‌های مطمئن، از جمله این چالش‌ها می‌باشد. به دلیل همین محدودیت‌ها پروتکل‌های مسیریابی برای این گونه شبکه‌ها به گونه‌ای طراحی می‌شوند که حداقل یکی از عوامل زبان بار شبکه را کمینه سازند. به عنوان مثال برخی از پروتکل‌ها صرفاً بر مبنای توان باتری گره‌ها بنا می‌شوند، بسیاری دیگر کمینه‌سازی سربارهای پردازشی را مورد توجه قرار می‌دهند و در بسیاری دیگر اجتناب از حلقه (که خود عامل بسیاری از محدودیت‌ها در شبکه از جمله کاهش پهنای باند و ... می‌باشد) هدف طراحی است. در این مقاله، ما به بررسی پروتکل SAODV و سطح امنیتی آن می‌پردازیم.

۲- مسیریابی چند مسیری در شبکه‌های موردی

این مسیریابی برای پیدا کردن چندین مسیر بین مبدأ و مقصد مفید می‌باشد. این مسیرهای چندگانه می‌توانند برای مقابله با تغییرات غیرقابل پیش‌بینی در شبکه‌های سیار موردی استفاده شوند. مسیریابی چند مسیری به دو صورت موجود است:

* مسیریابی با مسیر اصلی - فرعی [۱].۲

* مسیریابی چند مسیری^۲.

در مسیریابی با مسیر اصلی - فرعی چندین مسیر وجود دارند که فقط یکی از آن‌ها به عنوان مسیر اصلی استفاده می‌شود. در واقع مسیری به عنوان مسیر اصلی برای ارسال بسته‌های داده در نظر گرفته می‌شود و باقی مسیرها به عنوان مسیر فرعی و جایگزین می‌باشند. مسیریابی چند مسیری، چندین مسیر را برای ارسال داده به طور همزمان انتخاب می‌کند. در هر دو روش، کشف مجدد مسیرها زمانی انجام می‌شوند که کلیه مسیرهای موجود به سمت مقصد غیر قابل دسترسی باشند.

2- Primary-Secondary Routing

3- Multi path Routing

× نفوذگران با سرقت کدهای عبور و دیگر عناصر امنیتی، مشابه کاربران مجاز در شبکه‌های بی‌سیم، می‌توانند به شبکه مورد نظر بدون هیچ مانعی متصل گردند.

× با سرقت عناصر امنیتی، یک نفوذگر می‌تواند رفتار یک کاربر را پایش کند. از این طریق می‌توان به اطلاعات حساس دیگری نیز دست یافت.

× کامپیوترهای قابل حمل و جیبی، که امکان و اجازه استفاده از شبکه بی‌سیم را دارند، به راحتی قابل سرقت هستند. با سرقت چنین سخت‌افزارهایی، می‌توان اولین قدم برای نفوذ به شبکه را برداشت.

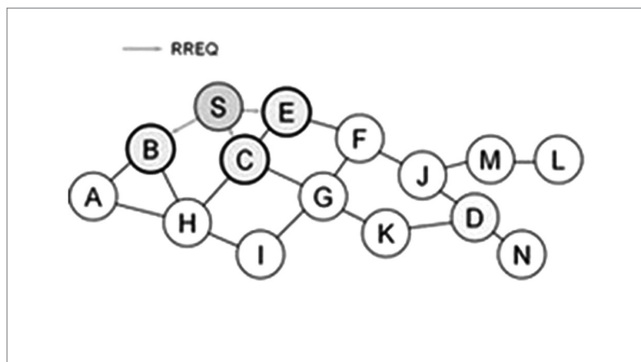
× یک نفوذگر می‌تواند از نقاط مشترک میان یک شبکه بی‌سیم در یک سازمان و شبکه سیمی آن (که در اغلب موارد شبکه اصلی و حساس‌تری محسوب می‌گردد) استفاده کرده و با نفوذ به شبکه بی‌سیم عملاً راهی برای دستیابی به منابع شبکه سیمی نیز بیابد.

× در سطحی دیگر، با نفوذ به عناصر کنترل‌کننده یک شبکه بی‌سیم، امکان ایجاد اختلال در عملکرد شبکه نیز وجود دارد.

در قسمت‌های آتی می‌توانیم به ویژگی‌های این شبکه‌ها، از بعد امنیتی بپردازیم و چگونگی پیکربندی صحیح یک شبکه بی‌سیم را، برای بالابردن امنیت آن، بررسی کنیم.

۷- AODV (Ad Hoc On demand Distance Vector)

AODV یک پروتکل واکنشی است که مبنای آن بر DSDV استوار است که در سال ۱۹۹۷ معرفی شده است. این پروتکل برای شبکه‌هایی با چند ده تا چند صد گره طراحی شده است. یکی از جنبه‌های AODV استفاده از شماره رشته (توالی) در جدول هر گره است. این شماره رشته توسط گره مقصد تولید می‌شود. شماره رشته، هم در بسته درخواست مسیر و هم در بسته‌های پاسخ به مسیر گنجانده می‌شود و به گره‌های درخواست‌کننده فرستاده می‌شود. شماره رشته از اهمیت فوق‌العاده‌ای برخوردار است، زیرا باعث اجتناب از حلقه شده و به سادگی قابل برنامه‌ریزی است. در دیگر گره‌ها نیز این شماره رشته به منظور به روزرسانی اطلاعات مسیریابی استفاده می‌شود. اگر یک گره بین دو مسیر حق انتخاب داشته باشد، مسیری را انتخاب می‌کند که شماره رشته آن بیشتر باشد. AODV با جداول مسیریابی سازگار است. بسته‌های درخواست مسیر، جواب به مسیر و خطا در AODV نیز همانند DSR تعریف می‌شود. مهمترین ویژگی این الگوریتم آن است که می‌تواند به صورت پویا خود را با تغییرات آرایش شبکه منطبق نماید، به طوری که با پایش کیفیت هر مسیر به مقصد، همیشه بهترین مقصد را مورد استفاده قرار دهد و مسیرهای قدیمی را حذف و مسیرهای کشف شده را نیز در صورت امکان کوتاه‌تر نماید و به این وسیله تعداد بسته‌های از دست رفته را کاهش دهد. الگوریتم مذکور برای شبکه‌هایی که مسیرهای چند مسیری را ارائه می‌دهند مناسب می‌باشد. در این روش، سربار کنترلی کمتری



شکل ۱

موارد ذکر شده نیازهای یک محیط باز^۴ را برآورده می‌کنند. دو مورد آخر مربوط به یک محیط باز مدیریت شده^۵ است. با این فرض که مدیریت شبکه در راهاندازی و توزیع کلید و ... نقش داشته باشد.

۶- منشأ ضعف امنیتی در شبکه‌های بی‌سیم و خطرات معمول

خطر معمول در کلیه شبکه‌های بی‌سیم مستقل از پروتکل و فناوری مورد نظر، بر مزیت اصلی این فناوری که همان پویایی ساختار، مبتنی بر استفاده از سیگنال‌های رادیویی به جای سیم و کابل است، استوار می‌باشد. با استفاده از این سیگنال‌ها و در واقع بدون مرز ساختن پوشش ساختار شبکه، نفوذگران قادرند در صورت شکستن موانع امنیتی نه چندان قدرتمند این شبکه‌ها، خود را به عنوان عضوی از این شبکه‌ها جا زده و در صورت تحقق این امر، امکان دستیابی به اطلاعات حیاتی، حمله به سرویس‌دهندگان سازمان و مجموعه، تخریب اطلاعات، ایجاد اختلال در ارتباطات گره‌های شبکه با یکدیگر، تولید داده‌های غیرواقعی و همراه کننده، سوءاستفاده از پهنای باند مؤثر شبکه و دیگر فعالیت‌های مخرب وجود دارد. در مجموع، در تمامی دسته‌های شبکه‌های بی‌سیم، از دید امنیتی حقایق مشترکی صادق است:

× تمامی ضعف‌های امنیتی موجود در شبکه‌های سیمی، در مورد شبکه‌های بی‌سیم نیز صدق می‌کند. در واقع نه تنها هیچ جنبه‌ای چه از لحاظ طراحی و چه از لحاظ ساختاری، خاص شبکه‌های بی‌سیم وجود ندارد که سطح بالاتری از امنیت منطقی را ایجاد کند، بلکه همان گونه که ذکر شد مخاطرات ویژه‌ای را نیز موجب می‌گردد.

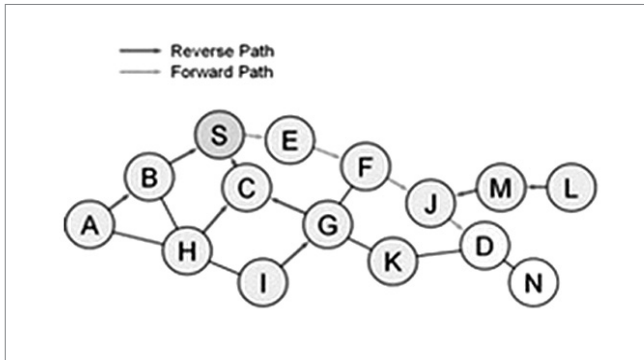
× نفوذگران، با گذر از تدابیر امنیتی موجود، می‌توانند به راحتی به منابع اطلاعاتی موجود بر روی سیستم‌های رایانه‌ای دست یابند.

× اطلاعات حیاتی که یا رمز نشده‌اند و یا با روشی با امنیت پایین رمز شده‌اند و میان دو گره در شبکه‌های بی‌سیم در حال انتقال می‌باشند، می‌توانند توسط نفوذگران سرقت شده یا تغییر یابند.

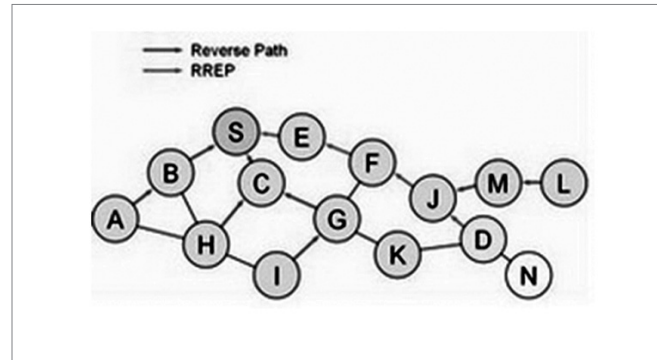
× حمله‌های DOS به تجهیزات و سیستم‌های بی‌سیم بسیار متداول است.

4- Open Environment

5- Managed Open Environment



شکل ۳



شکل ۲

برای اندازه‌گیری تعداد دستگاه‌هایی (نظیر مسیریاب‌ها) که بسته تاکنون طی کرده است، استفاده می‌گردد. اگر شمارش دستگاه‌های میان‌راهی از مقدار Max count بیشتر شود، این بسته نادیده گرفته می‌شود. برای عدم تغییر مقدار شمارش دستگاه‌های میان‌راهی و اطمینان از صحت مقدار آن، از توابع درهم‌ساز استفاده می‌شود. مثال زیر را برای شکل ۴ در نظر بگیرید.

فرض کنید در این شبکه گره S قصد پیدا کردن مسیری به سمت D را دارد. روند اجرای این پروتکل به صورت زیر است:

$S \rightarrow *:$ $\langle\langle RREQ, id, S, seq_S, D, oldseq_D, h_0, N \rangle_{K_S}, 0, h_N\rangle$
 $A \rightarrow *:$ $\langle\langle RREQ, id, S, seq_S, D, oldseq_D, h_0, N \rangle_{K_S}, 1, h_{N-1}\rangle$
 $B \rightarrow *:$ $\langle\langle RREQ, id, S, seq_S, D, oldseq_D, h_0, N \rangle_{K_S}, 2, h_{N-2}\rangle$
 $C \rightarrow *:$ $\langle\langle RREQ, id, S, seq_S, D, oldseq_D, h_0, N \rangle_{K_S}, 3, h_{N-3}\rangle$
 $D \rightarrow C:$ $\langle\langle RREP, D, seq_D, S, lifetime, h'_0, N \rangle_{K_D}, 0, h'_N\rangle$
 $C \rightarrow B:$ $\langle\langle RREP, D, seq_D, S, lifetime, h'_0, N \rangle_{K_D}, 1, h'_{N-1}\rangle$
 $B \rightarrow A:$ $\langle\langle RREP, D, seq_D, S, lifetime, h'_0, N \rangle_{K_D}, 2, h'_{N-2}\rangle$
 $A \rightarrow S:$ $\langle\langle RREP, D, seq_D, S, lifetime, h'_0, N \rangle_{K_D}, 3, h'_{N-3}\rangle$

همان‌طور که در بالا مشخص است، هر گره با دریافت یک پیام می‌تواند با کنترل $h_{n-1} = H(h_n)$ از اصالت آن مطمئن شود. عدد N نیز نشان‌دهنده حداکثر دستگاه‌هایی است که یک بسته می‌تواند طی کند [۵].

× جهت ایجاد امنیت در شبکه‌های موردی با استفاده از این الگوریتم می‌توان موارد زیر را اجرا نمود:

۱- استفاده از رمزنگاری کلید عمومی

۲- الحاق امضا به پروتکل مسیریابی AODV

۳- استفاده از پسوند تک امضایی RReq-SSE (RRep)

۴- احراز اصالت قسمت‌های (RRep)

۵- استفاده از زنجیره اعداد درهم (Hash Chains) برای احراز اصالت متریک (شمارش دستگاه‌های میان‌راهی)

۶- امکان استفاده از RRep-DSE و RReq-DSE

۷- هزینه محاسباتی بالا به دلیل استفاده از رمزنگاری نامتقارن

۹- مقایسه پروتکل‌ها از نظر کنترل گردش

در پروتکل DSR علیرغم این که پروتکلی واکنشی است به خاطر

نسبت به بقیه وجود دارد. شکل‌های ۱ و ۲ و ۳ نحوه عملکرد AODV را نشان می‌دهند. در وهله اول S می‌خواهد با D ارتباط برقرار کند، به همین منظور اقدام به انتشار بسته درخواست مسیر می‌کند تا مقصد را بیابد. این گره بسته درخواست مسیر را تولید می‌کند که شامل نشانی مقصد، شماره رشته و ID انتشار است، سپس آن را به همسایگانش ارسال می‌کند.

هر گرهی که این درخواست را دریافت می‌کند، مسیری در جهت عکس به گره ارسال می‌کند ضمن این که اگر به مقصد دسترسی نداشته باشد درخواست را به سمت دیگر همسایگانش به پیش می‌راند. مسیر زمانی یافت می‌شود که RREQ به گرهی برسد که به مقصد دسترسی دارد و زمانی مسیر قابل بهره‌برداری است که RREP از D به S برسد و در جدول مسیریابی S نگاه داشته شود. پس از دریافت RREP توسط هر گره، لازم است گره مورد نظر جدول مسیریابی خود را به روز کند (بر اساس شماره رشته).

هم‌اکنون S می‌تواند با D ارتباط برقرار کند.

زمانی که در یک مسیر فعال، پیوندی شکسته می‌شود یک پیام خطا (RERR) به دیگر گره‌ها ارسال می‌شود. اگر گره‌ها مسیری شامل پیوند شکسته شده در جدول مسیریابی خود دارند آن مسیر را حذف می‌کنند و گره S بار دیگر RREQ را به همسایگانش می‌فرستد. البته در این پروتکل هر گرهی که در مسیر قرار دارد می‌تواند مبادرت به یافتن مسیر به D کند. به چنین سازوکاری تعمیر محلی مسیر می‌گویند. اجتناب از حلقه، انتشار چندگانه اختیاری و کاهش سربارهای کنترلی از مزایای این پروتکل و تأخیر در فرآیند کشف مسیر و ضرورت وجود پیوندهای دوطرفه جهت آشکار شدن پیوندهای یک طرفه از جمله معایب آن است [۴].

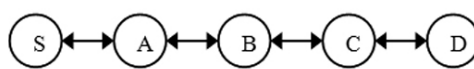
۸- پروتکل SAODV^۶

این پروتکل نیز برای ایجاد امنیت در الگوریتم AODV بنا شده است. در این پروتکل از توابع درهم‌ساز^۷ استفاده می‌شود، به طوری که $h_{n-1} = H(h_n)$ در این الگوریتم از شمارش دستگاه‌های میان‌راهی^۸

6- Secure AODV

7- hash function

8- hop count



شکل ۴. یک شبکه موردی نمونه

فراهم می‌کنند. ابزارهای پیشرفته‌تری همانند Kismet نیز بر بستر لینوکس معرفی شده‌اند. Kismet به صورتی نامحسوس (Pas-sive) ترافیک شبکه را ذخیره و پایش می‌کند. هر دوی این نرم‌افزارها، Netstumbler و Kismet، از اطلاعات GPS (سیستم مکان‌یابی جهانی) برای نگاشت مکان دقیق شبکه‌های محلی بی‌سیم استفاده می‌کنند. مهاجمان از این ابزار استفاده می‌کنند تا وجود فیزیکی شبکه بی‌سیم را تشخیص دهند، فارغ از این که این شبکه‌ها امن هستند یا خیر. جنگ‌گردان‌ها^{۱۰} کسانی هستند که با یک رایانه قابل حمل یا وسیله‌ای مشابه، داخل و اطراف شهرها می‌گردند تا سیگنال‌هایی از شبکه بی‌سیم بیابند. سپس این اطلاعات بر روی یک وبگاه قرار خواهد گرفت. رخنه‌گران از این اطلاعات و لیست‌ها استفاده می‌کنند تا نقاط دستیابی را با SSID نشانی Mac یا شماره فیزیکی مشترک در یک نشانی و موقعیت بیابند. آنتن‌ها برای اتصال به شبکه‌های محلی بی‌سیم از راه دور به کار می‌روند. رخنه‌گران یا از انواع بسیار متنوع آنتن‌های تجاری استفاده می‌کنند، یا این که به راحتی آنتن‌های خود را با قوطی خالی و یا هر وسیله مشابه فلزی دیگری می‌سازند. این آنتن‌ها رخنه‌گران را قادر می‌سازند تا امواج ۸۰۲٫۱۱ را از فاصله چند هزار متری دریافت کنند. آن‌ها می‌توانند به شبکه دسترسی داشته باشند در حالی که کاملاً دور از چشم همه قرار دارند.

ابزارهایی که رمزنگاری WEP را می‌شکنند

رنه‌گران از ابزاری همانند WEPwedge، WEPcrack، WEPAttack و BSD-Airtools و AirSnort برای شکستن رمزنگاری استاندارد WEP (Wired Equivalent Privacy) استفاده می‌کنند. این ابزارها از آسیب‌پذیری‌های موجود در الگوریتم رمزنگاری WEP استفاده می‌کنند، بدین شکل که به صورت کاملاً نامحسوس ترافیک شبکه محلی بی‌سیم را زیر نظر می‌گیرند تا زمانی که اطلاعات کافی برای تشخیص الگو به دست آورند. سپس از این اطلاعات برای شکستن کلید رمزنگاری استفاده می‌کنند. WEPwedge و BSD-Airtools زمان طولانی مورد نیاز برای شکستن کلیدهای بلند WEP را به حداقل می‌رسانند و این زمان را با استفاده از تکنیک تزریق ترافیک^{۱۱} از چند روز به چند ساعت تقلیل می‌دهند. در این روش حجم وسیعی ترافیک کاذب برای بازیابی کلید ایجاد می‌شود. معمولاً برای برپایی یک WEP دستی، اغلب تنها از یک کلید منفرد از چهار کلید برای گسترش شبکه استفاده می‌کنند که در مدت زمان بسیار کوتاه‌تری می‌توان شبکه را کاملاً تسخیر کرد. با وجود آسیب‌پذیری‌ها، WEP همچنان مورد استفاده قرار می‌گیرد. نسل جدید رمزنگاری‌ها از پروتکل TKIP استفاده می‌کند که امتیازاتی همچون به هم ریختن کلید در بسته، کنترل یکپارچگی و یک سازوکار کلیدبندی دوباره را فراهم می‌نماید. کلیدها به اندازه‌ای زود تغییر می‌کنند که مانع تسخیر و سوء استفاده شوند. اما چون اطلاعات روی هوا فرستاده می‌شود امکان دسترسی به

تعدد پیغام‌هایی نظیر RREQ، RREP، RERR و همچنین اطلاعاتی که گره‌ها به این پیغام‌ها اضافه می‌کند، حجم تبادل اطلاعات کنترلی در شبکه زیاد است. از طرفی AODV اساساً پروتکلی است برگرفته از DSDV که خود بر مبنای الگوریتم بردار فاصله عمل می‌کند. حال آن که DSR بر اساس موقعیت پیوند بنا نهاده شده است. لذا همین جا می‌توان نتیجه گرفت حجم اطلاعات کنترلی در شبکه هنگام استفاده از AODV کمتر از DSR است. البته اگر بررسی را با جزئیات بیشتری انجام دهیم، مشاهده می‌کنیم که طول قالب در بسته کنترلی AODV که مابین گره‌های واسط رد و بدل می‌شوند نیز از DSR کمتر است و این خود یعنی کمتر بودن سربارهای کنترلی در AODV نسبت به DSR. می‌دانیم هر چه اطلاعات گره‌ها راجع به موقعیت کلی شبکه بیشتر باشد نیاز به تبادل اطلاعات کنترلی کمتر پیش می‌آید پس مجهز نمودن گره‌ها به ادواتی چون GPS و استفاده از پروتکل‌های مبتنی بر موقعیت جغرافیایی می‌تواند سربارهای کنترلی را کاهش دهد. به علاوه می‌توان با به کارگیری پردازنده‌های قوی‌تر و تجهیزاتی چون آنتن‌های هوشمند این سربارها را به حداقل رساند. نهایتاً زمانی که پروتکل بر مبنای کنترل گردش بنا می‌شود از همان ابتدا کمینه‌سازی سربارهای کنترلی مهمترین هدف در انتخاب مسیر است، لذا محاسبات زمانی در گره‌ها لحاظ می‌شود [۶]. نتیجه کیفی مطالب فوق در شکل ۵ نشان داده شده است، جهت پیکان به سمت کاهش سربارهای کنترلی است. در نتیجه با توجه به حجم اطلاعات ردوبدل شده نسبت به حجم داده‌های موجود در قالب‌ها، سربارهای کنترلی پروتکل‌ها را به صورت کیفی با هم مقایسه نمودیم و نهایتاً به این نتیجه رسیدیم که تجهیز گره‌ها به ادواتی همچون آنتن‌های هوشمند و پردازنده‌های قوی می‌تواند به کاهش این سربارها کمک کند.

۱۰- چند راه‌حل برای برقراری امنیت شبکه‌های بی‌سیم

در این قسمت به معرفی چند ابزار می‌پردازیم که نمایانگر ناامنی و راه‌های سوء استفاده از شبکه‌های محلی بی‌سیم است. با دانستن این موارد مدیران شبکه بهتر می‌توانند در مورد امنیت شبکه خود تصمیم‌گیری کنند.

× ابزارهای Sniff و Scan شبکه‌های محلی بی‌سیم
ابزارهای مبتنی بر ویندوز، ساده و رایگانی همانند NetStumbler امواج هوا را scan کرده و با جستجوی نقاط دستیابی^۹ که Access ID خود را منتشر می‌کنند، راه بسیار ساده‌ای برای کشف شبکه‌های باز

10- war drivers

11- traffic insertion

9- access point



شکل ۵

مقابل آن ارائه می‌دهد و نه استفاده از شبکه‌های مجازی شخصی (VPN). احراز هویت لایه دوم شبکه‌های محلی بی‌سیم مبتنی بر ۸۰۲٫۱۱x، برای محافظت در مقابل ارتباطات مشکوک می‌تواند مفید باشد ولی دارای آسیب‌پذیری‌های زیادی است. یک ارتباط مغرضانه و مشکوک سعی در شکستن VPN و یا موازین امنیتی ندارد، ولی در عوض از لایه دوم برای تسلط بر کاربران استفاده می‌کند. برای جلوگیری از اتصال کاربرها به نقاط دسترسی و شبکه‌های غیرمجاز شرکت‌ها باید مرتباً امواج حوالی شبکه بی‌سیم خود را بررسی کنند تا از هرگونه خطر احتمالی مطلع شوند.

۱۱- نتیجه‌گیری

در این مقاله به بررسی مختصری بر امنیت مسیریابی در شبکه‌های موردی پرداختیم. با توجه به ضعف این گونه شبکه‌ها از نظر خطر در برابر حملات گوناگون جا دارد تا این ضعف‌ها در شبکه‌ها مورد بررسی دقیق‌تری قرار بگیرند تا با اطمینان بیشتری بتوان از آن‌ها استفاده نمود. شاید برخی از راه‌حل‌ها تعدادی از مشکلات امنیتی را برطرف کنند، اما ممکن است کارایی شبکه را دچار مشکل نمایند. بنابراین ارائه یک الگوریتم مسیریابی برای شبکه‌های موردی که هم از لحاظ امنیت و هم از لحاظ کارایی شبکه دارای سطح قابل قبولی باشد، ضروری به نظر می‌رسد.

۱۲- مراجع

- [1] R. Pearlman, J. Haas, P. Sholander, and S. Tabrizi, "On the Impact of Alternate Path Routing for Load Balancing in Mobile Ad Hoc Networks", IEEE Global Telecommunications Conference, PP 231-238 2000.
- [2] S. Mueller, P. Tsang, and D. Ghosal, "Multipath Routing in Mobile Ad Hoc Networks: Issues and Challenges", Springer COMMUNICATION WIRELESS NETWORK, PP 1-6, 2004.
- [3] Bridget Dahill et al, "A Secure Routing Protocol for Ad Hoc Networks," MobiCom 2002, Atlanta, Georgia, USA, September 23-28, 2002.
- [4] A. Young-Bae Ko and B. Nitin H. Vaidya., "Location Aided Routing (LAR) in mobile ad hoc network" 2004.
- [5] Po-Wah Yau and Chris J. Mitchell, "Reputation Methods for Routing Security for Mobile Ad Hoc Networks," Proceedings of SympoTIC '03 Joint IST Workshop on Mobile Future and Symposium on Trends in Communications, Bratislava, Slovakia, PP. 130-137, October 2003.
- [6] Sergio Marti et al, "Mitigating Routing Misbehavior in Mobile Ad Hoc Networks," Proceedings of the 6th annual international conference on Mobile computing and networking, Boston, USA, PP. 256-255, 2000.

آن وجود دارد و اگر رمزنگاری نشده باشد، به راحتی قابل استفاده خواهد بود.

ابزار شکستن احراز هویت

رخنه‌گران از ابزارهایی مانند THC-LEAPCracker برای شکستن یا تسخیر انواع مختلف و متداول پروتکل‌های احراز هویت مبتنی بر درگاه^{۱۲} برای ۸۰۲٫۱۱ بی‌سیم، مانند پروتکل^{۱۳} LEAP و یا^{۱۴} PEAP استفاده می‌کنند. این پروتکل‌ها برای استفاده شبکه‌های محلی با بستر سیمی که از نظر فیزیکی در محیطی امن قرار دارند طراحی شده‌اند. زمانی که اطلاعات در محیط اشتراکی و غیر قابل کنترل بی‌سیم پراکنده می‌گردد رخنه‌گران به راحتی می‌توانند گواهینامه‌های احراز هویت را spoof، jump in the middle و یا بوکشی کنند. حملات متداول شبکه‌های محلی بی‌سیم این قسمت چند حمله متداول بر روی شبکه‌های محلی بی‌سیم را بیان می‌کند که نمایانگر خطرات و ریسک‌های مشخص آن می‌باشد. با گوناگونی و تنوع فعلی ابزارهای رخنه‌گری که به صورت گسترده‌ای در اینترنت موجود می‌باشند حتی یک رخنه‌گر نوآموز و تازه‌کار می‌تواند بسیاری از حملات منتشر شده را اجرا نماید. تماس‌های تصادفی یا مغرضانه یک رخنه‌گر می‌تواند یک کاربر ساده را وادار نماید تا به صورت کاملاً ناخودآگاه به یک شبکه جعل شده ۸۰۲٫۱۱ متصل شود و یا این که تنظیمات این دستگاه را به گونه‌ای تغییر دهد که در یک شبکه اقتضایی قرار گیرد. برای شروع، رخنه‌گر از یک رایانه قابل حمل به عنوان یک نقطه دسترسی نرم‌افزاری استفاده می‌کند که برای این کار از ابزارهای رایگانی همچون HostAP، AirSnarf، Hotspotter و یا ابزارهای موجود تجاری می‌توان استفاده کرد. (به عنوان مثال شرکت‌هایی همچون PCTel نرم‌افزارهای تجاری تولید می‌کنند که تجهیزات ۸۰۲٫۱۱ را به نقطه دسترسی تبدیل می‌کند) همین‌طور که رایانه قربانی یک درخواست برای اتصال به یک نقطه دسترسی را منتشر می‌کند، نقطه دسترسی نرم‌افزاری رخنه‌گر به این درخواست پاسخ می‌دهد و یک اتصال بین این دو برقرار می‌گردد. سپس این نقطه دسترسی نرم‌افزاری یک نشانی IP به این رایانه اختصاص می‌دهد. پس از این که این کار انجام شد، رخنه‌گر می‌تواند رایانه قربانی را پویش کرده و در آن به گشت و گذار بپردازد، اطلاعاتی را برآید، اسب تروا و یا ابزار جاسوسی نصب کند و یا اگر رایانه قربانی به یک شبکه مبتنی بر سیم متصل باشد، از این رایانه می‌تواند به عنوان راه ارتباط برای نفوذ به کارسازهای دیگر این شبکه استفاده کند. شبکه‌های محلی بی‌سیم دستخوش دگرگونی فراوان هستند و اغلب ایستگاه‌های کاری نمی‌دانند به کدام نقطه دسترسی متصل هستند و از آنجایی که اغلب هیچ‌گونه احراز هویتی برای اتصال به نقاط دسترسی صورت نمی‌گیرد، ایستگاه‌های کاری می‌توانند فریب خورده و یا مجبور به اتصال با یک نقطه دسترسی ناامن شوند. این یک آسیب‌پذیری در لایه ۲ (Data Link) از مدل ۷ لایه شبکه OSI می‌باشد. نه احراز هویت لایه سوم هیچ‌گونه محافظتی در

12- port

13- Lightweight Extensible Authentication Protocol

14- Protected Extensible Authentication Protocol

برگزاری سمینار مروری بر جنبه‌های بارز ویرایش جدید C++

- C++: A definition (at 1998)
- The evolution of C++ programming language
- The C++0x core language features
- automatic type deduction - Initializers list and uniform initialization- range-based for loop – generalized constant expressions – R-value references and Move semantics - Defaulted and deleted functions – Lambda expressions - ...
- The C++0x standard library features
- New containers – New algorithms – Smart pointers– Tuples – Threads and concurrent programming facilities - ...
- C++: A definition (at 2011)

در پایان این سخنرانی که با حضور تعدادی از استادان دانشکده و تعداد زیادی از دانشجویان برگزار شد به پرسش‌های حاضران در مورد مسائل مطرح شده پاسخ داده شد.

لازم به ذکر است که آقای مهندس سعید امراللهی بیوکی، تنها عضو مشورتی کمیته استانداردسازی زبان C++ از ایران است که خود در آخرین نشست کمیته استانداردسازی این زبان حضور داشته است.

انجمن انفورماتیک ایران بدین وسیله از آقای مهندس امراللهی به خاطر قبول دعوت شاخه دانشجویی انجمن در دانشگاه تهران برای برگزاری این سخنرانی صمیمانه قدردانی می‌کند.

اسلایدهای این سخنرانی از طریق وبگاه انجمن (www.isi.org.ir) در اختیار علاقه‌مندان قرار دارد.



شاخه دانشجویی انجمن انفورماتیک ایران در دانشکده ریاضی، آمار و علوم کامپیوتر دانشگاه تهران یک سخنرانی علمی در تاریخ ۱۳۹۱/۲/۲۴ در محل سالن هشترومی این دانشکده برگزار کرد. سخنران این جلسه آقای مهندس سعید امراللهی بیوکی و موضوع سخنرانی، مروری بر جنبه‌های بارز ویرایش جدید زبان برنامه‌نویسی C++ بود.

آقای مهندس امراللهی در ابتدای سخنان خود با بیان این که زبان‌های برنامه‌نویسی نیز مانند زبان‌های طبیعی اگر می‌خواهند زنده بمانند باید تغییر کنند و با شرایط موجود وفق یابند، به ویژگی‌های ویرایش جدید زبان C++ به نام C++0x یا C++11 که توسط کمیته استانداردسازی زبان C++ تدوین شده پرداخت.

موضوعاتی که در این سخنرانی مورد بحث قرار گرفت به شرح زیر بود:

برگزاری سمینار صنعت بازی‌های رایانه‌ای



سخنرانی علمی اردیبهشت ماه شاخه دانشجویی انجمن انفورماتیک ایران در دانشکده مهندسی برق و کامپیوتر دانشگاه شهید بهشتی در تاریخ ۹۱/۲/۲۵ در محل دانشکده و در حضور دانشجویان مهندسی و علوم کامپیوتر آن دانشگاه برگزار شد. در این جلسه، آقای مهندس فرزاد ملک‌آرا درباره صنعت بازی‌های رایانه‌ای سخنرانی کرد.

آقای مهندس ملک‌آرا در ابتدای سخنان خود با اشاره به تعاریف مختلفی که از بازی رایانه‌ای به عمل آمده است، تعریف جامعی برای آن ارائه کرد و آنگاه به تفصیل به تشریح موضوعات زیر پرداخت:

- معرفی وضعیت صنعت بازی‌های رایانه‌ای در ایران
- زیرشاخه‌های مختلف صنعت بازی‌های رایانه‌ای
- تخصص‌های مورد نیاز برای تولید بازی
- فرصت‌های شغلی موجود در کشور
- مراحل ساخت بازی
- تفاوت‌های چرخه تولید نرم‌افزار و چرخه تولید بازی‌های رایانه‌ای
- مدیریت پروژه نرم‌افزاری در تقابل با مدیریت پروژه‌های بازی‌سازی

آقای مهندس ملک‌آرا در این سخنرانی که با استقبال زیادی از سوی حاضران روبرو گشت، چند کلیپ از بازی‌های رایانه‌ای معروف را نیز نشان داد و به تشریح ویژگی‌های آن‌ها پرداخت. در پایان سخنرانی، به پرسش‌های حاضران درباره موضوعات مطرح شده پاسخ داده شد.

انجمن انفورماتیک ایران بدین وسیله از آقای مهندس فرزاد ملک‌آرا، مدرس انستیتو ملی بازی‌سازی، به خاطر قبول دعوت شاخه دانشجویی

انجمن در دانشکده مهندسی برق و کامپیوتر دانشگاه شهید بهشتی برای برگزاری این سخنرانی صمیمانه قدردانی می‌کند.

انبار داده و پردازش تحلیلی برخط (OLAP)

داود صالحی

دانشکده مهندسی کامپیوتر دانشگاه شهید بهشتی

پست الکترونیکی: Goldsoft.salehi@gmail.com

چکیده

تحلیل داده‌های سازمان برای تصمیم‌گیری‌های آینده به موضوعی مهم و حیاتی برای سازمان‌های امروزی تبدیل شده است. در این راستا نقش انبارهای داده* و ابزارهای تحلیل داده‌ها جایگاهی ویژه یافته‌اند. برای تحلیل داده‌ها به منظور تصمیم‌گیری‌های آینده همواره نیازمند این هستیم تا داده‌های جاری و گذشته سازمان را که از منابع داده‌ای مختلف فراهم می‌شوند به صورت یکپارچه درآوریم و با استفاده از ابزارهایی این داده‌ها و اطلاعات را تحلیل نموده و به دانشی ارزشمند و قابل فهم برای کاربران نهایی تبدیل نماییم. لذا نیازمند ابزارهایی هستیم که امکان تعامل با داده‌های ذخیره شده در انبار داده را فراهم نموده و این داده‌ها را مورد پرس و جو** قرار داده و گزارش‌های مناسب را با توجه به نیازمندی‌های کاربر تهیه نمایند. همواره مجموعه‌ای از ابزارهای تحلیل داده وجود دارند که یکی از ابزارهای قدرتمند در این حوزه ابزارهای OLAP می‌باشد که داده‌های قرار گرفته شده در یک پایگاه داده چند بعدی را برای افزایش کارایی پرس و جوها و تهیه گزارش‌ها به کار می‌برد. کلمات کلیدی: انبار داده، OLAP، پایگاه‌های داده چند بعدی.

۱- مقدمه

سطح سازمان که همراه با تغییرات و بروز رسانی داده‌های عملیاتی سازمان همراه است مدیریت و کنترل نمایند. منظور از داده‌های عملیاتی، داده‌هایی است که یک سازمان در عملیات‌های روزانه خود به آن‌ها نیازمند است. این قبیل سیستم‌ها را که همواره در بردارنده تراکنش‌های سازمان و آن هم برای داده‌های عملیاتی سازمان می‌باشند سیستم‌های پردازش تراکنش برخط یا OLTP³ می‌نامند.

اما تحلیل‌گران و تصمیم‌گیرندگان یک سازمان برای بهبود کیفی

در یک نگاه کلی داده‌های مربوط به یک سازمان به دو دسته داده‌های عملیاتی^۱ و داده‌های تصمیم‌گیری^۲ تقسیم می‌شوند. اصولاً سازمان‌ها مدیریت داده‌های خود را از طریق سیستم‌های مدیریت پایگاه داده‌ای رابطه‌ای انجام می‌دهند. این سیستم‌ها برای این منظور طراحی شده‌اند که تراکنش‌های زیادی را در

* data warehouse

** query

1- operational data

2- decision data

3- OnLine Transaction Processing

یکپارچه، متغیر بر مبنای زمان^۶ و غیر فزّار که برای پشتیبانی فرایندهای تصمیم‌گیری مدیریتی استفاده می‌شود. در این تعریف خصیصه‌هایی برای داده در نظر گرفته شده است که عبارتند از [۲]:

- مبتنی بر موضوع: بدین معنا که انبار داده بر مبنای موضوع اصلی سازمان مورد نظر سازماندهی می‌شود. برای مثال این محور می‌تواند، مشتری، محصول و یا فروش باشد. برخلاف سیستم‌های پردازش تراکنش که مبتنی بر کاربرد می‌باشند، این خصیصه در نظر گرفته شده برای داده‌ها انعکاس دهنده این واقعیت است که در انبار داده‌ها نیارمند ذخیره‌سازی داده‌های تصمیم‌گیری می‌باشیم. برخلاف داده‌های مبتنی بر کاربرد که داده‌های عملیاتی سازمان و برای پشتیبانی از عملیات‌های روزانه سازمان به کار برده می‌شوند.
- یکپارچه: همان‌گونه که گفته شد، مجموعه داده‌های یک انبار داده از منابع مختلفی می‌آیند که اغلب شامل داده‌هایی می‌باشند که ناسازگار بوده و در قالب‌های مختلفی می‌باشند. منظور از ناسازگاری داده‌های منابع مختلف برای مثال وجود یک حوزه اطلاعاتی که در یک منبع وجود دارد و عدم وجود آن در منبع دیگری که تصمیم یکپارچگی آن‌ها را داریم می‌باشد. بنابراین یک منبع یکپارچه سازگار باید ساخته شود تا دید یکپارچه‌ای از داده‌ها به کاربران ارائه شود.

- متغیر بر مبنای زمان: داده‌های درون انبار داده در نقاطی از زمان و یا در بازه‌های زمانی مشخصی متغیر می‌باشند چرا که همواره داده‌های موجود در پایگاه داده عملیاتی و تراکنشی سازمان باید در بازه‌های زمانی خاصی به انبار داده انتقال پیدا کنند. بعد از انتقال عملیات تحلیل صورت می‌گیرد و بعد از گذشت مدتی تا زمانی که داده‌های جدیدی به درون انبار داده واکنشی نشده‌اند داده‌های موجود برای تصمیم‌گیری، داده‌های معتبری نخواهند بود [۳].

- غیر فزّار: بدین معناست که داده‌های ذخیره شده درون انبار داده به صورت بیدرنگ بروزرسانی نخواهند شد، بلکه تغییر محتویات درون انبار از سیستم‌های عملیاتی و در دوره‌های زمانی منظم صورت خواهد پذیرفت. داده‌های جدید همیشه به عنوان مکمل به پایگاه داده‌ها اضافه خواهند شد و جایگزین داده‌های قبلی نمی‌شوند و همواره به داده‌های قبلی الحاق می‌شوند.

تعریف ارائه شده برای انبار داده توسط اینمون یک تعریف متمرکز بر خصیصه‌های داده‌های ذخیره شده در انبار داده می‌باشد. تعاریف دیگری نیز برای انبار داده ارائه شده است که برای روشن شدن بهتر مفهوم انبار داده به آن اشاره خواهیم کرد. اما ذکر این نکته ضروری است که جدای از تعاریف متفاوتی که برای انبار داده ارائه شده است، هدف اصلی و نهایی انبار داده‌ها، یکپارچه‌سازی داده‌های سازمان درون یک مخزن به گونه‌ای که کاربران بتوانند به سادگی پرس و

سازمان خود نیازمند دسترسی به تمامی داده‌های سازمان، مستقل از محل قرارگیری آن‌ها می‌باشند، چراکه داده‌های عملیاتی سازمان می‌توانند تحت مدیریت سیستم‌های پایگاه داده‌ای متفاوت که از لحاظ فیزیکی نیز در یک مکان واقع نشده‌اند، باشند. از سوی دیگر برای فراهم کردن امکان تحلیل جامع سازمان مربوط و نیازمندی‌های آن‌ها، تصمیم‌گیرندگان و تحلیل‌گران، نه تنها نیازمند دسترسی به مقادیر جاری موجود در پایگاه داده می‌باشند، بلکه نیازمند دسترسی به داده‌های بایگانی شده و گذشته سازمان خواهند بود.

برای سهولت این قبیل تحلیل‌های سازمان در حیطه داده‌های وابسته به سازمان، انبارهای داده‌ای ایجاد شده‌اند که شامل داده‌هایی از منابع داده‌ای مختلف که توسط واحدهای عملیاتی مختلفی نیز نگهداری می‌شوند، می‌باشد [۱]. انبار داده بر مبنای فناوری توسعه یافته پایگاه داده مدیریت این قبیل داده‌های جمع‌آوری شده از منابع مختلف و مخازن داده‌ای را فراهم می‌آورد. هر چند در این بین برای تحلیل و فرآیندهای تصمیم‌گیری نیازمند ابزارهای تحلیل می‌باشیم. دو نوع اصلی ابزارهای تحلیل که جایگاه ویژه‌ای در این راستا دارند عبارتند از OLAP و ابزار های داده کاوی.

در ادامه ابتدا مروری بر ادبیات موضوع انبار داده خواهیم داشت و سپس مقایسه‌ای بین سیستم‌های پردازش تراکنش برخط و سیستم‌های انبار داده انجام داده و در ادامه معماری انبار داده و مولفه‌های تشکیل دهنده آن را مورد مطالعه قرار می‌دهیم. در پایان به معرفی ابزار OLAP خواهیم پرداخت.

۲- مفاهیم انبار داده و مروری بر نوشتگان موضوع

مفهوم اصلی پایگاه داده اولین بار توسط آی‌بی‌ام تحت عنوان «انبار اطلاعات»^۴ بیان شد و به عنوان یک راه حل برای دسترسی به داده‌هایی که درون سیستم‌های غیر رابطه‌ای نگهداری می‌شوند نشان داده شد [۱، ۲]. انبار اطلاعات بدین منظور پیشنهاد شد تا به سازمان‌ها این امکان را بدهد که از داده‌های آرشیو شده استفاده نموده تا به بهبود حرفه خود دست پیدا کنند. هر چند به دلیل پیچیدگی محض و مسایل مرتبط با کارایی، بسیاری از تلاش‌ها برای پیاده‌سازی راه‌حل‌های انبار اطلاعات رد می‌شد. بنابراین مفهوم انبار داده مطرح شد [۱]. آخرین و موفق‌ترین حمایت انجام شده برای انبار داده توسط بیل اینمون ارائه شده است [۱] و به او به دلیل فعالیت‌های انجام داده در ارتباط با مفهوم انبار داده لقب «پدر انبار داده» داده شده است [۱، ۲]. تعریفی که اینمون از انبار داده ارائه کرده است، در زیر بیان کرده ایم [۱، ۲]:

انبار داده عبارت است از مجموعه‌ای از داده‌های مبتنی بر موضوع^۵

4- information warehouse

5-subject-driven

۲-۲- جدول داده‌های واقعی

یکی از عناصر کلیدی یک انبار داده یا مخزن داده جدول داده‌های واقعی است. منظور از آن، جدولی است که شامل داده‌های واقعی حرفه مورد نظر که انبار داده برای آن ساخته می‌شود، می‌باشد. برای مثالی از داده‌های یک جدول داده‌های واقعی می‌توان تعداد مشتریان، میزان فروش، تعداد فروش و... را نام برد. همواره از این جدول یک کلید خارجی به جدول ابعاد وجود دارد [۲].

۲-۳- جدول ابعاد

جدولی که اطلاعات مربوط به یک قلم اطلاعاتی را دارد که کلید خارجی آن در جدول داده‌های واقعی موجود بوده و همواره شامل اطلاعات مربوط به تحلیل در یکی از ابعاد به عنوان مثال بعد زمان برای بررسی میزان فروش در زمان خاص می‌باشد [۲].

۲-۴- مکعب

یک مکعب یک ساختار داده چند بعدی است که برای اطلاعات تجاری مورد پرس و جو قرار می‌گیرد. در واقع مکعب یک پایگاه داده چند بعدی است که داده‌ها درون سلول ذخیره می‌شوند و موقعیت هر سلول توسط تعدادی متغیر که بعد نامیده می‌شوند، مشخص خواهد شد. هر سلول یک رویداد حرفه را نشان می‌دهد و مقادیر هر بعد مشخص می‌کنند که این رویداد در چه زمانی و در چه موقعیتی رخ داده است. نمونه‌ای از بعد می‌تواند زمان، مشتری و انبار باشد که مقدار درون یک سلول مکعب مشخص می‌کند که یک مشتری، کالای خاصی را در زمان خاصی از انبار خریداری نموده است [۳]. شکل این مکعب با ابعاد گفته شده را در زیر مشاهده می‌کنید.

۳- مقایسه سیستم‌های OLTP و انبار داده

همان‌گونه که توصیف آن رفت، همواره یک انبار داده، داده‌های خود را از منابع داده‌ای OLTP دریافت می‌کند و همواره حجم وسیعی از داده‌ها را در خود دارد. اصولاً سیستم‌های OLTP برای انبار داده مناسب نمی‌باشند، زیرا هر یک با مجموعه متفاوتی از نیازمندی‌ها ساخته شده‌اند و این سیستم‌ها بدین منظور طراحی شده‌اند تا ظرفیت پردازش تراکنش را حداکثر نمایند به گونه‌ای که باید در این گونه سیستم‌ها توانایی انجام چند صد تراکنش در ثانیه وجود داشته باشد. اما سیستم‌های انبار داده برای پشتیبانی از پرس و جوهای موردی^{۱۲} و فاقد عمومیت طراحی می‌شوند. یک سازمان ممکن است تعداد متفاوتی سیستم‌های OLTP داشته باشد که هر یک از سیستم‌های مجزا برای فرآیندهای حرفه خاص مورد استفاده قرار می‌گیرد. برای مثال یک بانک ممکن است یک برنامه کاربردی برای پردازش وام، یک سیستم برای سرویس مشتریان و یک سیستم جداگانه برای سرویس خودپرداز داشته باشد. این سیستم‌ها

جوی‌های خود را اجرا نموده، گزارش‌هایی را تولید کرده و فرآیند تحلیل را انجام دهند می‌باشد [۱].

اما تعریف جامع دیگری که برای انبار داده ارائه شده است عبارت است از [۳]:

انبار داده سیستمی است که داده‌ها را از سیستم‌های مبدأ^۷ به صورت دوره‌ای استخراج و ادغام نموده و آن‌ها را درون مخزن داده‌های نرمال شده^۸ و یا چند بعدی^۹ قرار می‌دهد و اصولاً داده‌های گذشته و چندین سال مربوط به سازمان را در خود نگهداری می‌کند که برای اهداف هوش تجاری و فعالیت‌های تحلیلی مورد پرس و جو واقع می‌شود. فرآیند بهنگام سازی یک انبار داده، یک فرآیند دسته‌ای^{۱۰} است و بهنگام سازی آن در زمانی که یک تراکنش در سیستم مبدأ به وقوع می‌پیوندد انجام نخواهد پذیرفت.

۲-۱- مخزن داده^{۱۱}

مفهوم دیگری که در ارتباط با انبار داده وجود دارد، مخزن داده می‌باشد. به زیرمجموعه‌ای از انبار داده که نیازمندی‌های یک بخش یا واحد خاص را پشتیبانی می‌کند مخزن داده گفته می‌شود [۱،۳،۵]. به عبارت دیگر مخزن داده به عنوان یک بخش از یک انبار داده بزرگ عمل خواهد نمود [۲]. یک مخزن داده می‌تواند مستقل باشد و یا این که به انبار داده سازمان پیوند داشته باشد. اما هدف از مخزن داده به عنوان یک زیر مجموعه از انبار داده این است که در اثر رشد انبار داده و بزرگ شدن آن، توانایی سرویس دهی به نیازمندی‌های مختلف یک سازمان در معرض خطر واقع می‌شود، لذا به ساخت مخزن داده خواهیم پرداخت. چراکه ساخت و استفاده از انبار داده‌های وسیع بسیار مشکل است.

ویژگی‌هایی که انبار داده و مخزن داده را از یکدیگر متمایز می‌کند عبارت است از [۱]:

- تمرکز مخزن داده فقط بر نیازمندی‌های کاربران خاص وابسته به یک واحد و یا کارکرد حرفه خاص می‌باشد.
- مخزن داده برخلاف انبار داده فاقد داده‌های عملیاتی تفصیلی می‌باشد.
- مخزن داده‌ها در قیاس با انبار داده‌ها حاوی اطلاعات کمتری می‌باشد اما به سادگی قابل درک و راهبری خواهد بود.
- معماری مخزن داده‌ها می‌تواند بر مبنای یک معماری دو لایه‌ای یا سه لایه‌ای برنامه‌های کاربردی پایگاه داده ساخته شود. در معماری سه لایه اگر انبار داده فراهم کننده داده‌های مخزن داده باشد، می‌تواند به عنوان اولین لایه، مخزن داده به عنوان لایه دوم و ایستگاه‌های کاری کاربر که از ابزارهای تحلیل مانند OLAP استفاده می‌کنند در لایه سوم قرار گیرد.

7-source systems

8-normalized

9-dimensional

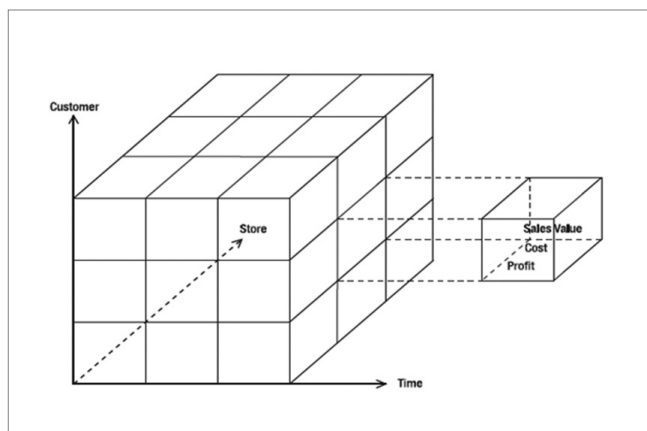
10-batch

11-Data mart

12-ad hoc

جدول ۱: مقایسه سیستم‌های انبار داده با سیستم‌های پردازش تراکنش‌های برخط

سیستم‌های OLTP	سیستم‌های انبار داده
نگهداری داده‌های جاری	نگهداری داده‌های بایگانی
ذخیره داده‌های تفصیلی	ذخیره داده‌های تفصیلی، داده‌های خلاصه و داده‌های توابع جمعی اعمال شده بر آن‌ها
داده‌ها پویا هستند	داده‌ها ایستا هستند
توان عملیاتی بالای تراکنش	توان عملیاتی سطح پایین یا سطح متوسط برای پردازش تراکنش
مبتنی بر تراکنش	مبتنی بر تحلیل
پشتیبانی از تصمیم‌گیری‌های روزانه	پشتیبانی از تصمیم‌گیری‌های راهبردی
سرویس دهی به تعداد زیادی کاربران عملیاتی	سرویس دهی به تعداد کمی از کاربران مدیریتی



شکل ۱: یک مکعب با سه بعد

RMS نگهداری می‌شوند و یا تحت مدیریت پایگاه داده‌های رابطه‌ای از قبیل اوراکل و یا اینفورمیکس واقع شده‌اند.
 - داده‌های خصوصی نگهداری شده در ایستگاه‌های کاری و کارسازهای (سرویسگر) خصوصی.

- سیستم‌های خارجی از قبیل اینترنت، پایگاه‌های تجاری موجود و یا پایگاه‌های وابسته به مشتریان و یا تولیدکنندگان یک سازمان.

- **مدیر بارگیری:** تمامی عملیات وابسته به استخراج و بارگیری داده‌ها به درون انبار داده توسط این مولفه انجام می‌پذیرد. این عملیات شامل تبدیل ساده داده‌ها برای آماده‌سازی آن‌ها برای ورود به انبار داده می‌باشد. پیچیدگی این مولفه بین انبار داده‌ها متفاوت است و می‌تواند از ترکیب ابزارهای بارگیری داده فروشندگان مختلف و یا برنامه‌های ساخته شده سفارشی ایجاد شود.

- **مدیر انبار داده:** مدیر انبار تمامی عملیات وابسته به مدیریت داده‌ها را درون انبار داده انجام خواهد داد. این عملیات شامل موارد زیر می‌باشد:

- تحلیل داده‌ها برای اطمینان از سازگاری داده‌ها.

داده‌های عملیاتی تفصیلی و مقادیر جاری را برای یک موجودیت خاص اخذ می‌کنند. سازماندهی داده‌ها در سیستم OLTP مطابق با نیازمندی‌های تراکنش‌های وابسته به برنامه‌های کاربردی حرفه و پشتیبانی از تصمیم‌گیری‌های روزانه و بر اساس تعداد کاربران جاری عملیاتی، انجام می‌پذیرد. اما یک سازمان به صورت طبیعی دارای یک انبار داده خواهد بود که داده‌های آرشیو شده، با جزئیات کافی و خلاصه شده در سطوح مختلف را نگهداری می‌کند. اصولاً این سیستم‌ها برای پشتیبانی از تعداد کمی تراکنش طراحی می‌شوند و سازماندهی داده‌های آن‌ها بر اساس نیازمندی‌های پرس و جوهای خاص و پشتیبانی از تصمیم‌گیری‌های راهبردی بلند مدت و برای تعداد کمی از کاربران مدیریتی انجام خواهد پذیرفت.

اگرچه سیستم‌های انبار داده و OLTP تفاوت‌های فاحشی با یکدیگر دارند و هریک برای منظور خاصی نیز طراحی می‌شوند، اما ارتباط نزدیکی با یکدیگر دارند به گونه‌ای که سیستم‌های OLTP همواره فراهم کننده داده‌های منبع برای انبار داده خواهند بود. اما مسئله اصلی در این ارتباط این است که داده‌های نگهداری شده توسط سیستم‌های OLTP می‌توانند در حالت ناسازگار، گسسته و شامل ورودی‌های تکرار شونده و یا از دست رفته باشند. لذا قبل از این که داده‌های عملیاتی درون انبار داده بار شوند، باید پالایش و ادغام شوند. پالایش داده‌های عملیاتی قبل از بار شدن آن‌ها در انبار داده را در بخش معماری انبار داده توصیف خواهیم کرد.

با توجه به موارد ذکر شده تفاوت‌های دو سیستم OLTP و انبار داده در جدول ۱ خلاصه شده است.

۴- معماری انبار داده

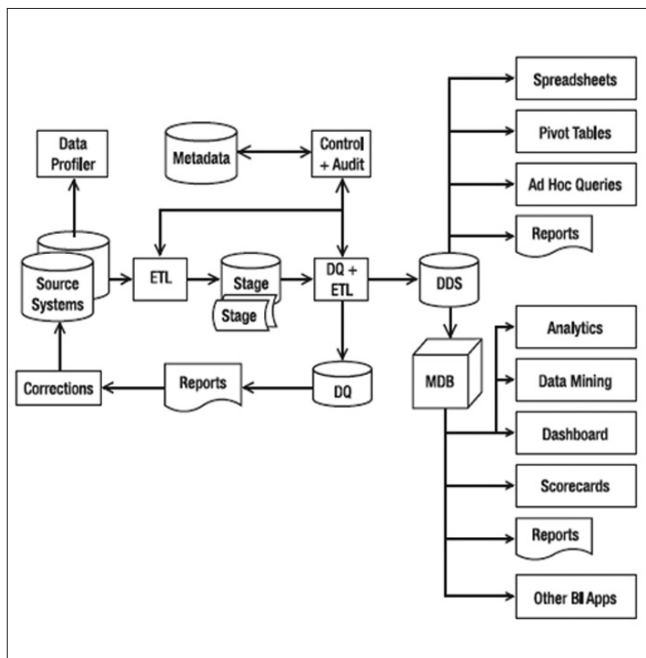
در این بخش به بررسی معماری سیستم انبار داده خواهیم پرداخت و مولفه‌های اصلی یک انبار داده را مرور خواهیم نمود. شکل مربوط به معماری عمومی یک انبار داده در ذیل مشاهده می‌شود [۱].

در این شکل مجموعه‌ای از مولفه‌ها مشهود می‌باشند که هر یک نقش خاصی را ایفا نموده و از تعامل آن‌ها یک سیستم انبار داده به وجود خواهد آمد. هر یک از این مولفه‌ها را مورد بررسی قرار می‌دهیم.

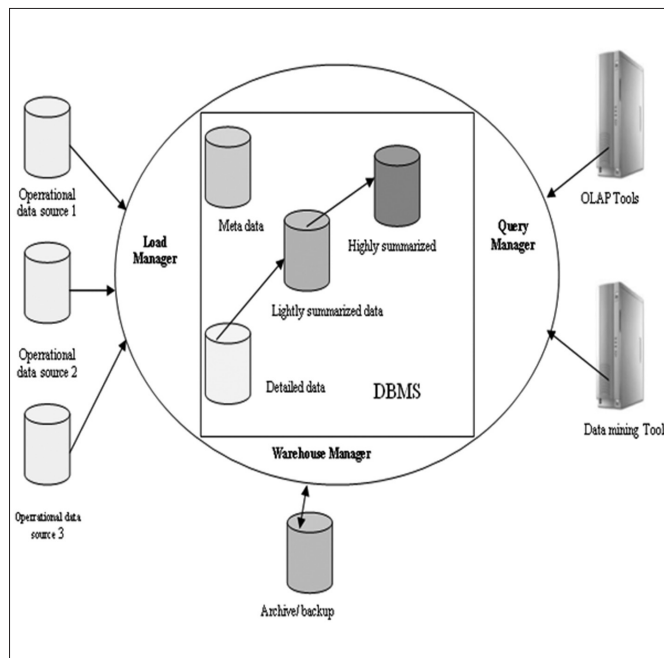
- **داده‌های عملیاتی:** همان گونه که پیش‌تر گفته شد، داده‌های مورد استفاده سیستم‌های OLTP که توسط تراکنش‌های روزانه و برای تصمیم‌گیری‌های مرتبط با عملیات روزانه سازمان مورد پردازش قرار می‌گیرند، داده‌های عملیاتی نامیده می‌شوند. این داده‌ها به عنوان یک منبع انبار داده و فراهم کننده داده‌های یک انبار می‌باشند. اما از لحاظ این که این داده‌ها از چه سکو و چه ساختار پایگاه داده‌ای بیایند می‌توانیم دسته‌بندی زیر را برای آن‌ها در نظر بگیریم:

- داده‌های عملیاتی یک رایانه بزرگ که در پایگاه داده سلسله مراتبی و شبکه‌ای نگهداری می‌شوند.

- داده‌های یک واحد که درون سیستم پرونده‌هایی از قبیل VSAM و



شکل ۳: معماری با تفصیلی سیستم انبار داده



شکل ۴: معماری عمومی انبار داده

داده به درون انبار داده بار شده‌اند اما هنوز توابع جمعی برای خلاصه‌سازی این داده‌ها بر روی آن‌ها اعمال نشده است) به صورت بخط ذخیره نمی‌شوند اما با سطح بعدی جزئیات یکپارچه می‌شوند.

- **داده‌های تلخیصی:** این ناحیه از انبار تمامی داده‌های متراکم شده و تجمیع^{۱۴} شده توسط مدیر انبار را ذخیره می‌کنند. هدف از تجمیع و خلاصه‌سازی داده‌ها و اطلاعات سرعت در کارایی پرس و جوها می‌باشد.
- **داده‌های بایگانی / پشتیبانی:** این ناحیه از انبار داده‌ها، داده‌های خلاصه شده و داده‌های تفصیلی را برای اهداف بایگانی و پشتیبان ذخیره می‌کند. هر چند داده‌های خلاصه شده از داده‌های تفصیلی تولید می‌شوند اما اگر این داده‌ها فراتر از دوره بقای داده‌های تفصیلی نگهداری شوند، عملیات پشتیبان‌گیری از داده‌های خلاصه شده ضروری است.
- **فراداده^{۱۵}:** تمامی فراداده‌های تعریف شده که توسط تمامی فرایندها در انبار استفاده می‌شوند، در این مولفه ذخیره می‌شوند.
- **ابزارهای دستیابی کاربر نهایی:** هدف پایه‌ای یک انبار داده فراهم کردن اطلاعات برای کاربران حرفه برای تصمیم‌گیری‌های راهبردی می‌باشد. برای تعامل با انبار داده، کاربران از ابزارهای دسترسی استفاده می‌نمایند. تعریف‌های ارائه شده برای ابزارهای دستیابی کاربران نهایی به نوعی روی هم افتادگی دارند. اما این ابزارها می‌توانند به پنج گروه اصلی تقسیم شوند [۱]:

- ابزارهای گزارش‌گیری و پرس و جو
- ابزارهای توسعه کاربردها

- انتقال و ادغام کردن داده‌های منبع از مخزن موقت به جداول انبار داده.

- ایجاد دیدگاه‌ها^{۱۳} و نمایه‌ها بر روی جداول پایه.

- اعمال فرآیند نرمال‌زدایی در صورت نیاز. (در توضیح باید افزود که داده‌های درون انبار داده برخلاف سیستم‌های OLTP نرمال نمی‌باشند. زیرا نرمال بودن داده‌ها باعث طولانی شدن زمان الحاق جداول در زمان اجرای پرس و جو می‌شود و از آنجایی که انبار داده حجم وسیعی از داده‌ها را دارد و به مراتب بزرگتر از یک پایگاه داده عملیاتی است نیازمند فرایند نرمال‌زدایی می‌باشیم. البته این در حالتی است که از پایگاه‌های داده چند بعدی استفاده نکنیم)

- اعمال فرایند یکپارچه‌سازی در صورت نیاز

- بایگانی کردن داده‌ها و گرفتن نسخه پشتیبان از آن‌ها.

در برخی موارد این مولفه یک شرح وضعیت پرس و جو ایجاد می‌کند تا مشخص کند چه نمایه‌ها و یکپارچه‌سازی‌هایی مناسب می‌باشند.

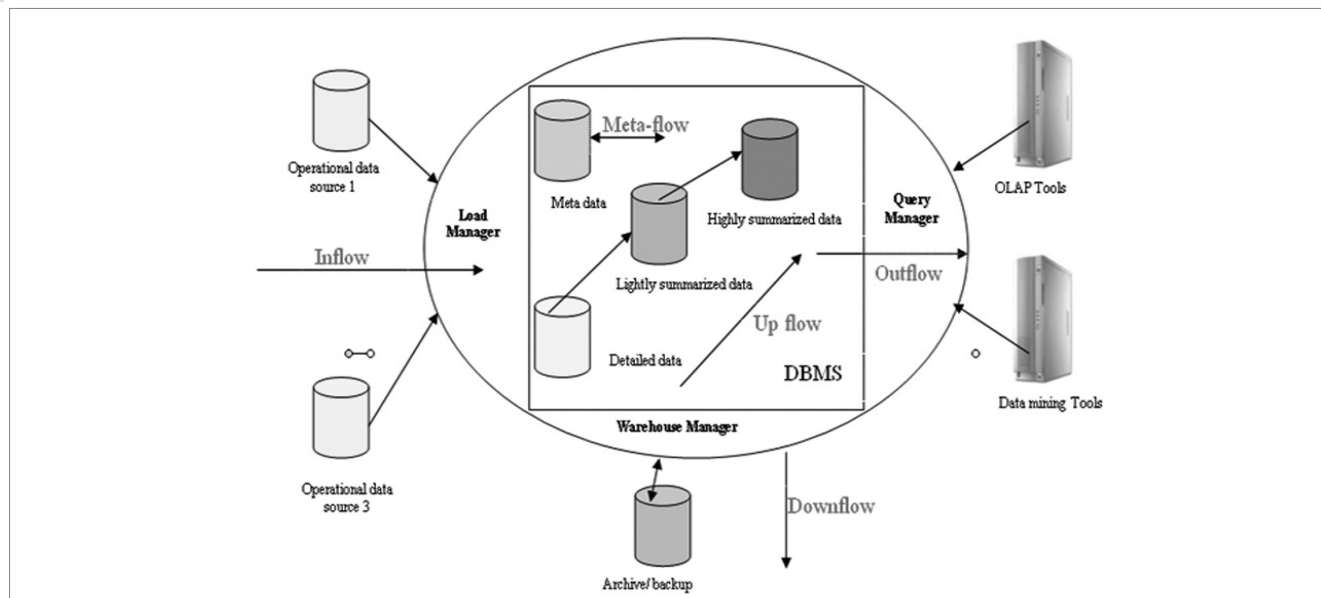
• **مدیر پرس و جو:** تمامی عملیات وابسته به مدیریت پرس و جوهای مورد نیاز کاربران توسط این مولفه انجام می‌پذیرد. این مولفه عموماً از ابزارهای دسترسی به داده کاربران نهایی، ابزارهای پایش انبار داده و ... ساخته می‌شود. در بسیاری از موارد شرح وضعیت پرس و جو توسط این مولفه ایجاد می‌شود تا به مدیر انبار اجازه دهد که مشخص کند که چه نمایه‌ها و یکپارچه‌سازی‌هایی مناسب می‌باشند.

• **داده‌های تفصیلی:** این ناحیه از انبار تمامی داده‌های تفصیلی درون شمای پایگاه داده را ذخیره می‌کند. در بسیاری از موارد، داده‌های تفصیلی (منظور داده‌های خامی می‌باشد که از منابع

14- Aggregate data

15-Meta data

13-view



شکل ۴: جریان اطلاعات یک انبار داده

که داده‌های درون انبار داده را به قالب‌های متفاوتی از سیستم OLTP ذخیره می‌کند. دلیل استفاده از DDS برای ذخیره داده‌ها و پرس و جو قرار دادن DDS به جای پرس و جوی منابع داده به صورت مستقیم این است که داده‌ها درون DDS به قالب مرتب شده‌اند که این قالب چند بعدی برای تحلیل داده‌ها بسیار مناسب است [۳]. دلیل دیگر استفاده از DDS این است که DDS شامل داده‌های یکپارچه شده از چندین سیستم منبع می‌باشد [۲، ۵].

مولفه دیگری که در این معماری وجود دارد، مولفه کیفیت داده‌ها می‌باشد. در زمان بار کردن داده‌ها به درون DDS قوانین کیفی داده‌ها، مجموعه‌ای از شرایط کیفی را بررسی می‌کند. داده‌هایی که این قوانین کیفی را نقض می‌کنند، درون پایگاه داده DQ قرار می‌گیرند و این موارد گزارش داده شده و سپس درون سیستم‌های مبدأ اصلاح خواهد شد. سیستم ETL توسط سیستم کنترل، مدیریت و هم‌نوا سازی می‌شود. این مدیریت بر مبنای قوانین، دنباله و منطق‌های ذخیره شده در پایگاه داده فراداده‌ها صورت خواهد پذیرفت. مولفه audit یا ممیزی که با سیستم کنترل ادغام شده است، عملیات سیستم را ثبت نموده و درون پایگاه داده فراداده‌ها استفاده می‌شود. سیستم ممیزی بخشی از سیستم ETL است که فعالیت‌های عملیاتی ETL را پردازش نموده و آمارهای عملیاتی را ثبت می‌کند. این سیستم برای درک این که در طول فرایند ETL چه اتفاقی افتاده است استفاده می‌شود.

برای تحلیل داده‌های قرار گرفته شده در DDS همواره مجموعه‌ای از ابزارها مانند کاربرگ‌ها^{۱۷}، جداول pivot یا مفصل، ابزارهای گزارش‌گیری و ابزارهای پرس و جوی SQL مورد استفاده قرار می‌گیرند. اما مجموعه‌ای از برنامه‌های کاربردی وجود دارند که بر روی پایگاه داده‌های چند بعدی عملیات را انجام می‌دهند. برای این

- ابزارهای سیستم اطلاعات اجرایی (EIS)

- ابزارهای OLAP

- ابزارهای داده‌کاوی

معماری ارائه شده قبلی به صورت یک معماری سطح بالا برای سیستم‌های انبار داده‌ای بود. اما برای مشاهده یک معماری دقیق‌تر و با جزئیات بیشتر شکل ۳ را در نظر گرفته و با مولفه‌های اساسی آن بیشتر آشنا می‌شویم [۳].

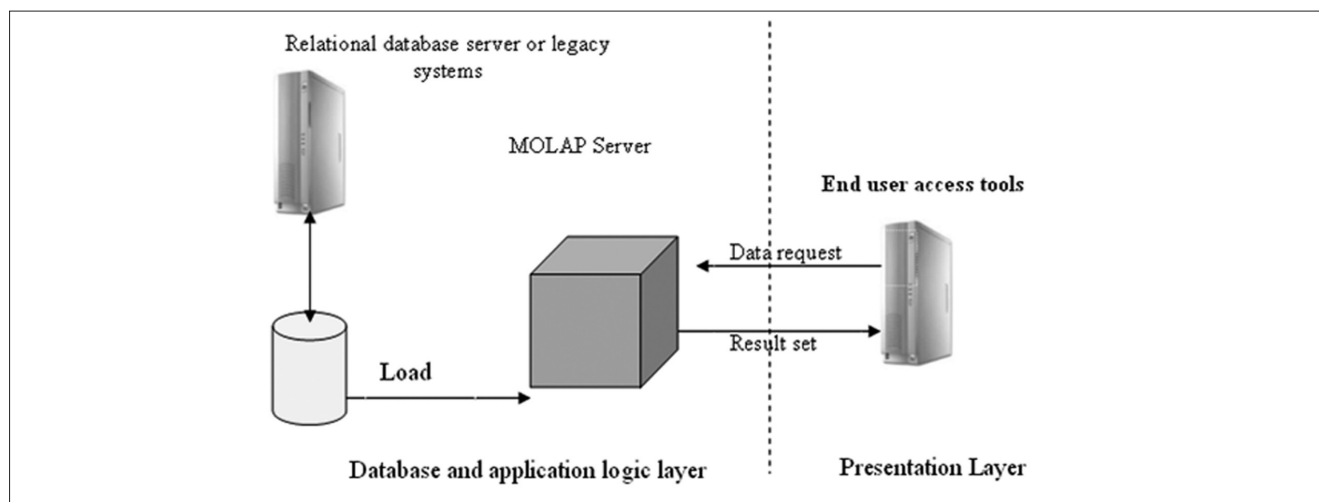
مولفه‌های سیستم مبدأ و فراداده در معماری شکل ۲ توصیف شدند. اما مجموعه‌ای از این مولفه‌ها در این شکل مشاهده می‌شود که در معماری ارائه شده در شکل ۲ وجود نداشتند. البته معماری ارائه شده در شکل ۲ معماری جامعی است، اما در عین حال یک معماری با جزئیات کم می‌باشد. معماری ارائه شده در شکل ۳ یک معماری تفصیلی می‌باشد.

مولفه ثبت شرح وضعیت داده‌ها ابزاری است که توانایی تحلیل داده‌ها از قبیل یافتن تعداد سطرهای موجود در جداول و این که چه تعداد از ردیف‌ها شامل مقدار null می‌باشند و مواردی از این دست را دارا می‌باشد. سیستم‌های مبدأ که منطبق بر مولفه مبدأ داده‌های عملیاتی در معماری ارائه شده در شکل ۲ می‌باشد از این ابزار استفاده نموده تا خصیصه‌های داده‌ها را درک نماید. سیستم ETL که خلاصه شده «استخراج-تبدیل-بار کردن» می‌باشد، داده‌ها را از منابع مختلف دریافت نموده و آن‌ها را به ناحیه stage انتقال می‌دهد. سیستم ETL توانایی اتصال به منابع مختلف، خواندن داده‌ها، تبدیل آن‌ها و در نهایت بار کردن داده‌های خوانده شده به سیستم هدف را دارا می‌باشد. سیستم ETL منطبق بر مولفه مدیر بارگیری در معماری ارائه شده در شکل ۲ می‌باشد.

مولفه ذخیره‌گاه داده‌های چند بعدی^{۱۶} (DDS) یک پایگاه داده است

17- spreadsheets

16- Dimensional data store



شکل ۵: معماری MOLAP

می‌باشد، بالاخص اگر این پایگاه داده در سطح سه نرمال و یا بیشتر باشد به دلیل عدم وجود افزونگی یا حداقل افزونگی عملیات پردازش تراکنش به منظور بهنگام‌سازی داده‌ها با کارایی مناسب صورت می‌پذیرد [۳].

۵- جریان اطلاعات^{۱۸} انبار داده

همواره در طول ایجاد و استفاده از یک انبار داده، از مرحله بارگیری داده‌ها از منابع داده‌های عملیاتی تا لحظه تحلیل داده‌ها توسط ابزارهای موجود، فعالیت‌های وابسته به جریان اطلاعات یا پردازش آن‌ها درون انبار داده رخ می‌دهد. در این بخش تمرکز ما بر روی این جریان‌های اطلاعات می‌باشد. انبار داده متمرکز بر مدیریت پنج جریان اطلاعات اصلی می‌باشد که این جریان‌ها عبارتند از: inflow, Upflow, Downflow, outflow, meta-flow. جریان اطلاعات در یک انبار داده با توجه به شکل شماره ۲ که نشانگر یک معماری سطح بالای سیستم می‌بود، در شکل شماره ۴ نمایش داده شده است:

اما هر یک از این جریان‌های اطلاعات شامل فرایندهای متفاوتی می‌باشند که در ذیل به پردازش‌های وابسته به هر یک از این جریان اطلاعات می‌پردازیم.

۵-۱- Inflow

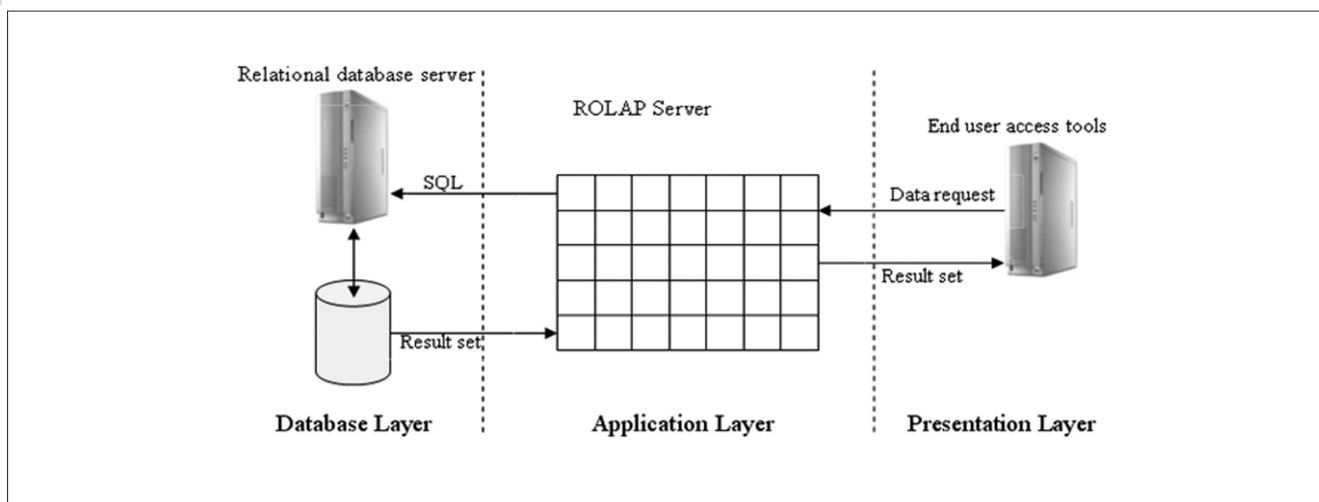
پردازش‌های وابسته و مرتبط با واکنشی، پالایش، ادغام و بارگیری داده‌ها از سیستم‌های مبدأ به درون انبار داده در این جریان اطلاعات انجام می‌پذیرد. پالایش داده‌ها بدین منظور صورت می‌گیرد که اطمینان حاصل شود داده‌ها کیفیت مطلوب برای بار شدن در انبار داده را داشته باشند. به عبارت بهتر داده‌ها باید برای اهداف انبار داده بازسازی دوباره شوند. این دوباره بازسازی شامل موارد زیر است [۱،۳]:

- پاکسازی داده‌های کثیف و ناجور

قبیل برنامه‌های کاربردی داده‌های درون DDS به درون یک پایگاه داده چند بعدی "MDBs" بار می‌شوند. به پایگاه داده‌های چند بعدی مکعب نیز گفته می‌شود [۴] که تعریف آن را پیش‌تر ارائه نمودیم. ذکر این نکته ضروری است که سیستم‌های انبار داده، بعد از بازیابی داده‌ها از منابع داده توسط ETL آن‌ها را فقط درون یک مخزن داده چند بعدی قرار نخواهند داد، بلکه این داده‌ها می‌توانند درون یک DDS یا مخزن داده چند بعدی و یا درون یک مخزن داده نرمال قرار گیرند. بنابراین برخی از انبارهای داده دارای قالب چند بعدی بوده و برخی دارای قالب نرمال می‌باشند [۱،۳،۵]. یک مخزن داده دارای بعد یا چند بعدی غیر نرمال می‌باشد و ابعاد در این پایگاه داده هم‌منوا و منطبق هستند. منظور از ابعاد منطبق و هم‌منوا این است که هر یک از ابعاد دقیقاً یک جدول می‌باشند و یا این که یک بعد زیرمجموعه‌ای از دیگری است [۳]. گفته می‌شود بعد A زیر مجموعه بعد B است اگر تمامی ستون‌های بعد A در بعد B و تمامی سطرهای بعد A در بعد B موجود باشد [۳].

یک مخزن داده دارای بعد می‌تواند از نظر فیزیکی به صورت‌های مختلفی از شمایها و الگوها پیاده‌سازی شود. نمونه‌ای از شمایهای مربوط به مخزن داده چند بعدی عبارت است از: snowflake schema, star schema, galaxy schema [۲،۳]. اما مخزن داده نرمال یک یا چند پایگاه داده رابطه‌ای با حداقل افزونگی می‌باشد و یک پایگاه داده رابطه‌ای یک پایگاه داده شامل جداول موجودیت با ارتباط پدر/فرزندی بین آن‌ها است [۱،۴]. نرمال‌سازی فرآیند از بین بردن افزونگی داده‌ها توسط پیاده‌سازی قوانین نرمال سازی می‌باشد.

اما تفاوتی که بین دو قالب برای انبار داده وجود دارد، مسئله کارایی در پرس و جو می‌باشد. یک مخزن داده چند بعدی و غیر نرمال یک قالب مناسب برای اهداف پرس و جو و تحلیل داده‌ها نسبت به یک پایگاه داده نرمال می‌باشد. از طرفی یک پایگاه داده نرمال قالب مناسبی برای یکپارچه‌سازی داده‌ها از منابع داده‌ای مختلف



شکل ۶: معماری ROLAP

۳-۵ Downflow

پردازش‌های وابسته و مرتبط با بایگانی، پشتیبان‌گیری و بازیابی داده‌ها درون انبار داده‌ها در این گردش اطلاعات صورت می‌پذیرد. بایگانی داده‌های قدیمی نقش مهمی در نگهداشت کارایی انبار توسط انتقال داده‌های قدیمی‌تر به مخزن بایگانی از قبیل دیسک نوری و یا نوار مغناطیسی بازی می‌کند. زیرا با این کار از رشد غیر قابل کنترل انبار داده‌ها جلوگیری کرده و نگهداشت و مدیریت آن با سهولت بیشتری انجام خواهد پذیرفت. هر چند اگر الگوی بخش‌بندی مناسبی برای پایگاه داده انتخاب شود، میزان داده‌های برخط بر روی کارایی تاثیر گذار نخواهد بود.

۴-۵ Outflow

پردازش‌های مرتبط با در دسترس قرار دادن داده‌ها برای کاربران نهایی در این جریان اطلاعات انجام خواهد پذیرفت. دو فعالیت کلیدی در این جریان انجام می‌پذیرد که عبارت است از:

- دستیابی: که مرتبط با برآورده ساختن درخواست‌های کاربر نهایی برای داده‌هایی که آن‌ها نیاز دارند می‌باشد. بحث اصلی در این فعالیت ایجاد محیطی است که کاربران بتوانند به صورت کارآمد از ابزارهای پرس و جو برای دسترسی به داده‌ها استفاده نمایند.
- حمل: مرتبط با فعالیت‌های تحویل اطلاعات به ایستگاه‌های کاری کاربران نهایی می‌باشد. این فعالیت حوزه نسبتاً جدیدی در انبار داده‌ها محسوب می‌شود و به عنوان فرایند publisher/subscriber نامیده می‌شود.

۵-۵ Meta-flow

فرایندها و پردازش‌های وابسته با مدیریت فراداده‌ها در این گردش اطلاعات انجام خواهد پذیرفت. این فرایندی است که فراداده را انتقال می‌دهد. به عبارت دیگر داده‌هایی در ارتباط با دیگر جریان‌ها می‌باشد. برای پاسخگویی به تغییرات نیازمندی‌های حرفه، سیستم‌های موروثی

- بازسازی دوباره داده‌ها برای تناسب با نیازمندی‌های جدید انبار داده. برای مثال نرمال‌زدایی داده‌ها و یا افزودن و حذف حوزه‌ها
- اطمینان از این که داده‌های منبع در حالت سازگار می‌باشند و همچنین اطمینان از این که سازگاری داده‌های قبلی در انبار داده برقرار است.

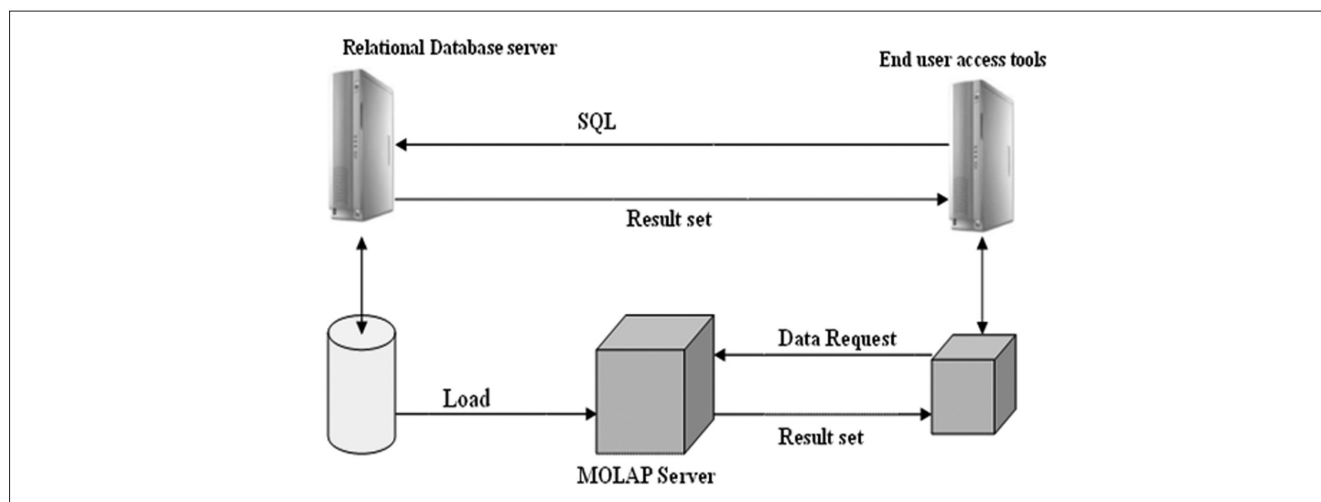
برای مدیریت کارآمد inflow سازوکارهایی باید تعریف شود تا مشخص کند چه هنگامی باید فرایند واکنشی داده‌ها صورت پذیرد [۱]. بعد از بازیابی داده‌ها، داده‌ها معمولاً در یک مخزن موقت قرار می‌گیرند تا عملیات پالایش و بررسی سازگاری داده‌ها اعمال شود. به دلیل پیچیدگی این فرایند، مهم است که این فرایند به صورت تمام خودکار بوده و توانایی اعلام گزارش در زمان وقوع خطا وجود داشته باشد.

۲-۵ Upflow

منظور از upflow همان گونه که از شکل نیز پیداست، پردازش‌های مرتبط با افزودن مقادیر به داده‌ها درون انبار داده، از طریق خلاصه‌سازی، بسته‌بندی و توزیع داده‌ها می‌باشد. فعالیت انجام شده در این گردش اطلاعات به قرار زیر است:

- خلاصه سازی^{۱۹}: خلاصه سازی داده‌ها توسط عملیات select, proj, grouping, join, ect داده‌های رابطه‌ای درون دیدگاه‌ها که برای کاربران نهایی مفید و مناسب می‌باشند.
- بسته بندی^{۲۰}: بسته‌بندی داده‌ها توسط تبدیل داده‌های تفصیلی یا خلاصه شده به قالب‌های مناسب از قبیل کاربرگ‌ها، مستندات متنی، نمودارها یا نمایش گرافیکی.
- توزیع^{۲۱}: توزیع داده‌ها در گروه‌های مناسب برای افزایش دسترسی پذیری داده‌ها.

19- summarizing
20-packaging
21-Distributing



شکل ۷: معماری ابزارهای MQE

نمونه‌ای از تحلیل‌ها که توسط OLAP انجام می‌پذیرد، تحلیل افزایش فروش در نواحی متفاوت و گروه‌های محصولات در یک دوره از زمان می‌باشد.

OLAP همواره تعاملی است [۲،۳] و ابزارهای OLAP ابزارهایی تعاملی می‌باشند که تحلیل‌گران حرفه‌ای از این ابزارها استفاده نموده تا انبار داده را مورد پرس و جو قرار داده و تحلیل‌های حرفه‌ای سازمان را انجام دهند. کارکردهای OLAP می‌توانند با استفاده از پایگاه‌های رابطه‌ای و یا با استفاده از پایگاه‌های چند بعدی تحویل داده شوند.

۶-۲ OLAP چند بعدی

یک OLAP که از پایگاه داده‌های چند بعدی استفاده می‌کند را به عنوان OLAP چند بعدی می‌شناسیم [۵] به عبارت دیگر سرویس‌دهنده‌های پایگاه داده‌های OLAP که از یک ساختار چند بعدی برای ذخیره داده‌ها و ارتباط بین آن‌ها استفاده می‌کنند را OLAP چند بعدی می‌نامیم [۱]. ساختار چند بعدی بهترین تصور از یک مکعب از داده‌ها و همچنین مکعب‌هایی که درون مکعب‌های دیگر واقع شده‌اند می‌باشد. هر چند برای پرس و جوها می‌توانیم از جداول دو بعدی و جداول رابطه‌ای استفاده نماییم اما عملیات مربوط به خلاصه‌سازی داده‌ها و اعمال توابع جمعی بر روی حجم وسیعی از داده‌ها در پایگاه داده‌های رابطه‌ای فرایندی به شدت زمانگیر است. اما سیستم مدیریت پایگاه داده‌های چند بعدی می‌تواند توابع جمعی مانند گروه‌بندی و خلاصه‌سازی را در زمانی معقول انجام دهد.

سرویس‌دهنده‌های پایگاه‌های داده‌ای OLAP چند بعدی عملیات تحلیلی متداول زیر را پشتیبانی می‌کنند:

- ادغام [۲]: یکپارچه‌سازی داده‌ها از قبیل انباشتن‌های ساده یا عبارات پیچیده درگیر با داده‌های وابسته به هم.

دائماً تغییر می‌کنند. بنابراین مدیریت انبار داده درگیر پاسخگویی به این تغییرات پیوسته می‌باشد. در این حالت meta-flow نیز باید به صورت پیوسته با تغییرات بهنگام شود.

۶-۳ پردازش تحلیلی برخط

تا بدین جا، سیستم‌های انبار داده، معماری این سیستم‌ها و جریان اطلاعات و داده‌های این سیستم‌ها را مورد بررسی قرار دادیم و آموختیم که یک سیستم انبار داده حجم وسیعی از داده‌های جاری و گذشته یک سازمان را در خود ذخیره نموده است که این داده‌ها برای تحلیل به منظور تصمیم‌گیری‌های آینده بهینه شده‌اند.

اما برای تحلیل داده‌های موجود در انبار داده‌ها همواره نیازمند ابزارهایی می‌باشیم تا بتوانیم گزارش‌های آماری مناسبی را ایجاد کنیم. در این راستا توسعه‌های کلیدی در ابزارهای دسترسی کاربر نهایی به انبارهای داده‌ای صورت پذیرفته است. یکی از این توسعه‌ها شامل پردازش تحلیلی برخط یا OLAP می‌باشد. دیگر ابزارها در این راستا شامل توسعه SQL برای تحلیل داده‌های پیچیده و ابزارهای داده‌کاوی می‌باشند.

۶-۱ پردازش تحلیلی برخط

پردازش تحلیلی برخط یا OLAP یک ترکیب پویا از تحلیل و ادغام حجم وسیعی از داده‌های چند بعدی می‌باشد [۱]. OLAP واژه‌ای است که ابتدا توسط کاد مطرح شده و معماری است که کاربردهای تحلیلی پیچیده را پشتیبانی می‌کند. برنامه‌های کاربردی OLAP با استفاده از فناوری چند بعدی DBMS، مجموعه محدودی از داده‌ها و واسطه‌های کاربری سفارشی شده پیاده‌سازی می‌شوند. تعریف دیگری که از OLAP در [۳] ارائه شده عبارت است از:

فعالیت‌های تعاملی تحلیل تراکنش‌های مخزن داده‌ها در انبار داده چند بعدی برای تصمیم‌گیری‌های راهبردی و تاکتیکی حرفه.

22- Multidimensional OLAP

23-Consolidation

محصولات ROLAP دارای یک موتور SQL توسعه یافته می‌باشند که پیچیدگی تحلیل چند بعدی را پشتیبانی می‌کنند. معماری عمومی برای سیستم‌های ROLAP در شکل ۶ قابل مشاهده است.

۳-۳-۶ - MQE^{۲۴}

این ابزارها جدیداً توسعه داده شده‌اند و توانمندی‌های تحلیلی محدودی را به صورت مستقیم از طریق RDBMS و یا با استفاده از سرویس دهنده‌های MOLAP میانی فراهم می‌کنند. ابزارهای MQE داده‌های انتخاب شده را به صورت مستقیم از DBMS ها یا از طریق سرویس دهنده‌های MOLAP به فرم مکعب‌های داده‌ها تحویل می‌دهند و به صورت محلی ذخیره، تحلیل و نگهداری خواهند شد. معماری این ابزار را در شکل شماره ۷ ملاحظه می‌کنید.

۴-۳-۶ - HOLAP

همان‌گونه که گفته شد سیستم‌های HOLAP از هر دوی MOLAP و ROALP استفاده می‌کنند. زمانی که از SQL Server Analysis Service برای ذخیره داده‌ها در حالت HOLAP استفاده می‌کنیم، داده‌های در سطح برگ درون جداول پایگاه داده رابطه‌ای ذخیره می‌شوند و داده‌های انباشت شده و خلاصه شده درون پایگاه داده چند بعدی - MDB - ذخیره خواهند شد [۲،۳].

۲-۲-۶ - زبان پرس و جوی OLAP

SQL زبان استاندارد پرس و جو برای پایگاه‌های داده رابطه‌ای می‌باشد. به غیر از ویژگی‌های اندکی از OLAP که به SQL-99 افزوده شده است، زبان مشابهی برای تحلیل وجود ندارد. اما میکروسافت عبارت‌های چند بعدی MDX^{۲۵} را ارائه نموده است که یک استاندارد غیررسمی دارد. MDX یک زبان برنامه نویسی است که به عنوان بخشی از OLAP+SQL Server می‌باشد. قواعد نحوی این زبان کمی گیج‌کننده بوده و ترکیبی از SQL و گویش شیء‌گرا می‌باشد. این زبان در ارتباط با بسته‌های کامل آماری مانند SAS کمی ضعیف‌تر می‌باشد [۲،۵].

مراجع

- [1] Connolly Thomas and Begg Carolyn, Database Systems A practical approach to Design, Implementation and Management, 3rd ed., Addison Wesley, 1999.
- [2] C. Matt, M. Sethu, Z. Robert, G. Denny, Microsoft SQL Server Analysis Services 2008 with MDX, 1rd ed., Wrox, Canada, 2009.
- [3] Rainardi Vincent, Building a Data Warehouse with example in SQL Server, 2rd ed., Apress, 2008.
- [4] Silberschatz, Korth, Database Systems Concepts, 4rd ed., 2004.
- [5] Celko's Joe, Analytics and OLAP in SQL, Morgan Kaufmann, 2009.

• کل به جزء: عملیاتی بر عکس ادغام را انجام می‌دهد و داده‌های تفصیلی را نمایش می‌دهد.

• خرد کردن: این عملیات که مفصل‌بندی نیز نامیده می‌شود، اشاره به توانایی مشاهده داده‌ها از دیدگاه‌های متفاوت دارد. خرد کردن در محور زمان انجام می‌پذیرد تا گرایش‌های تحلیل و الگوها یافته شوند.

سرویس‌دهنده‌های OLAP توانایی ذخیره داده‌های چند بعدی در قالب فشرده شده را دارا می‌باشند و این کار توسط انتخاب پویای مخزن‌های فیزیکی سازمان‌ها و تکنیک‌های فشرده‌سازی انجام خواهد پذیرفت. داده‌ها با چگالی بالا از داده‌ها با فشرده‌گی کم و پراکنده به صورت جداگانه ذخیره خواهند شد.

۳-۳-۶ - دسته‌بندی ابزارهای OLAP

ابزارهای OLAP بر مبنای معماری پایگاه داده قرار گرفته شده در زیر آن‌ها تقسیم‌بندی می‌شوند. سه دسته بندی اصلی از ابزارهای OLAP وجود دارد که عبارتند از [۱]:

MOLAP یا OLAP چند بعدی که MD-OLAP نیز نامیده می‌شوند. ROLAP یا OLAP رابطه‌ای و MQE یا محیط پرس و جوی مدیریت شده.

نوع دیگری از ابزارهای OLAP وجود دارد که آن را HOLAP یا OLAP دورگه می‌نامند که از هر دو پایگاه داده رابطه‌ای و چند بعدی استفاده می‌کند [۳،۵].

۱-۳-۶ - MOLAP

ابزارهای MOLAP از ساختارهای داده‌ای خاص و سیستم‌های مدیریت پایگاه داده چند بعدی MDDBMS برای سازمان‌دهی، راهبری و تحلیل داده‌ها استفاده می‌کنند. ساختار داده‌های MOLAP از فناوری آرایه و تکنیک‌های ذخیره‌سازی کارآمد برای کاهش نیازمندی‌های فضای دیسک و مدیریت داده‌های پراکنده استفاده می‌نمایند. ابزارهای MOLAP کارایی بسیار خوبی را فراهم می‌کنند و یک اتصال محکم با لایه کاربرد و لایه نمایش دارند. به عنوان مثال‌هایی از ابزارهای MOLAP می‌توان Oracle's express serv-er, Pilot software Analysis server, Sinper's TM/1 را نام برد. معماری سیستم MOLAP در شکل ۵ مشاهده می‌شود.

۲-۳-۶ - ROLAP

OLAP رابطه‌ای یک فناوری OLAP می‌باشد که به سرعت در حال رشد می‌باشد. ROLAP محصولات سیستم‌های مدیریت پایگاه داده‌های رابطه‌ای را پشتیبانی می‌کند و این کار را از طریق استفاده از لایه فراداده انجام می‌دهد. بنابراین نیازمند ایجاد ساختار داده‌ای چند بعدی ایستا نمی‌باشیم. این کار سبب سهولت ایجاد دیدگاه‌های چند بعدی از رابطه‌های دو بعدی می‌شود [۱]. یکی دیگر از مزایای ROLAP این است که داده‌های تحلیلی و پایگاه رابطه‌ای از یک موتور برای ذخیره و بازیابی استفاده می‌نمایند [۵]. برای بهبود کارایی، برخی از

جهان رقمی آینده: قواعد معماری، معیارهای مدیریت پروژه و آداب فناوری و مهندسی

سید ابراهیم ابطاحی

عضو هیأت علمی دانشکده مهندسی کامپیوتر - دانشگاه صنعتی شریف

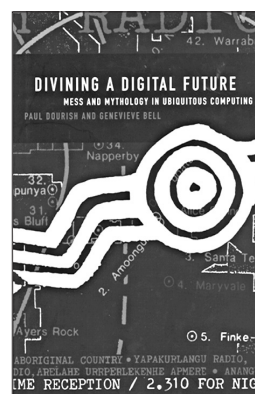
پست الکترونیکی: abtahi@sharif.edu

بیست و پنجمین نمایشگاه بین‌المللی کتاب تهران به روال هر سال در بهار امسال از ۱۲ تا ۲۲ اردیبهشت ۱۳۹۱ در مصلای تهران برگزار گردید. از تکرار مکررات تبدیل تدریجی این نمایشگاه به بازار کتاب می‌گذریم که به نظر می‌رسد کاملاً محقق شده است که در این صورت چه بهتر که از این پس به شکل دائمی و روزآمد در خدمت شهروندان باشد.

از ویژگی‌های این دوره به غیر حضور انبوه بازدیدکنندگان و تعداد زیاد ناشران کمتر شناخته شده با تعداد قلیل عناوین منتشره روزآمد و غیبت تعدادی از ناشرین معتبر بود. تعداد قلیل عناوین معتبر کتاب‌های خارجی دانشگاهی در حوزه رایانه و فناوری اطلاعات بیش از گذشته ناامیدکننده بود، آن هم برای رشته‌ای با این همه موضوعات و کتاب‌های روزآمد منتشره. ضمن ارج‌گذاری به نیت فرهنگی برنامه‌ریزان و زحمات اجرایی مجریان، به نظر می‌رسد پسندیده باشد مسئولین این نمایشگاه برای تفاهم بیشتر با علاقه‌مندان کتاب، ضمن ترسیم شفاف اهداف برگزاری این نمایشگاه، سالیانه گزارشی از هزینه‌ها و منفعتهای آن را که از سوی خبرگان غیر درگیر اجرای نمایشگاه تهیه شده منتشر نمایند تا با قلت کاهنده تعداد عناوین و شمارگان کتب ارزشمند منتشره، ضرورت برگزاری این گردهمایی فرهنگی را به چالش نکشد. در ادامه به رسم هر ساله، برای علاقه‌مندان موضوعات مرتبط با رایانه و فناوری اطلاعات به معرفی چهار کتاب منتخب و عرضه شده در این نمایشگاه می‌پردازیم.

مقدمه

تبعات آینده رقمی در راه، متأثر از عصر سوم فناوری اطلاعات، زمانه رایانش حی و حاضر - در دسترس همه جا و هر زمان - (UbiComp : Ubiquitous Computing) در پی رایانش فراگیر (Pervasive Computing) موضوع این کتاب خواندنی منتشر شده توسط موسسه فناوری ماساچوست است. گروه بزرگی از پژوهشگران از دانشگاه‌های معتبر و مراکز پژوهشی به نام، همکاران نویسندگان این کتاب بوده‌اند که علاوه بر یادآوری



عنوان کتاب: Divining a Digital

Future Mess and Mythology in Ubiquitous Computing

نویسندگان: Paul Dourish

and Genevieve Bell

ناشر: The MIT Press

زمان نشر: ۲۰۱۱

تعداد صفحات: 248 Pages

شابک: 978-0-262-01555-4

مقدمه

معماری سازمانی از موضوعات مهم و اصلی دهه گذشته و آینده فناوری اطلاعات) بوده و خواهد بود. الزام یکپارچه سازی ضرورت معماری فکورانه و اتصال پذیر همه مولفه ها (فناوری ها، داده ها، کاربردها، سامانه ها و فرآیندها) را آشکار ساخته و آن را به عنوان پیش نیاز اجرای پروژه های جهانی نظیر برپائی دولت های الکترونیکی در جایگاه رفیعی قرار داده است. پیچیدگی های اجرائی مطالعات معماری به عنوان مطالعاتی راهبردی، در میان دو طیف ریز بین عملگرا و کلی نگر الگو گرا در نحله ای به سوی اصول گرایان مقید به قواعد با اجرای موردی معماران- که سبک آنان را می سازد- میل کرده است که این کتاب نمونه ای آغازین از این گرایش در معماری می تواند تلقی شود. بر همین مبنا است که نویسندگان این کتاب عنوان دوم آن را اساس معماری سازمان نام نهاده اند.

در این کتاب گریف هورست و پراپر نویسندگان آن بر نقش این قواعد تاکید کرده اند و نگاهی دوسویه واجد هر دو جنبه نظری و عملی در گزینش این قواعد و اصول داشته اند. در نگرش نظری به بررسی اجمالی بیشینه مفاهیم مطرح از دیدگاه های متفاوت پرداخته اند و معیار آن ها بررسی نظریاتی بوده است که راه به کاربرد و به کارگیری و رد پا در بهترین تجارب اجرائی هم داشته اند. از بعد عملی، آن ها این نظریات را تا اجرا پیگیری و به کمک این بررسی ها، سبیدی قابل اعتنا از اصول معماری را گرد آورده اند که به این کتاب ارزشی خاص داده است. سید و کاتاکوگ ۵۹ قاعده ای گرد آمده و درج آن در پیوست اول کتاب نشانه توفیق نویسندگان در تحقق اهدافشان در نگارش این کتاب است. نویسندگانی که شرح حال تخصصی آن ها در پایان کتاب نشانه تبحرشان در این حوزه هاست. پراپر که از مدیران گروه تخصصی Archimate در Open Group است و گریف هورست که عضو هیئت مدیره دپارتمان معماری انجمن رایانه آلمان است.

محتوای کتاب

کتاب دارای یک پیشگفتار، مقدمه ای چهار بخشی، هفت فصل، دو پیوست مفصل، واژه نامه، فهرست منابع و بخش معرفی نویسندگان است. عناوین فصول کتاب به شرح زیر است:

- ۱- فصل یک: مقدمه
 - ۲- فصل دو: نقش معماری سازمانی.
 - ۳- فصل سه: چارچوبی مفهومی برای اصول.
 - ۴- فصل چهار: تشریحات اصول معماری.
 - ۵- فصل پنج: یک مشی عملی.
 - ۶- فصل شش: موارد مطالعاتی.
 - ۷- فصل هفت: در متن اصول معماری.
 - ۸- فصل هشت: خلاصه، نتیجه گیری و کارهای آتی.
- نگارنده که منتقد کیفیت برخی از پایان نامه های تحصیلی

نویسندگان، صورت غنی منابع کتاب یادآور اعتبار و ارزش های پژوهشی مطالب مطروحه کتاب است.

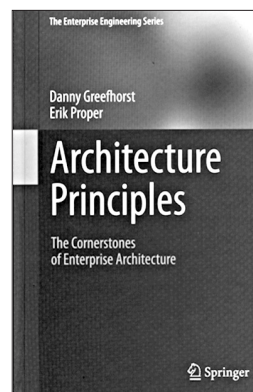
واکوی این گونه رایانش از نگاهی میان و فرارشته ای- به قصد گذر از اسطوره سازی های اغراق آمیز و یا اشاره به دشواری های سطحی عیان- عمق پژوهشی مطالب مطروحه را باز می نماید. این نگاه همه جانبه است که نویسندگان را به طرح مباحثی می کشاند که بسیار تازه است از جمله بندی در فصل هفتم کتاب با عنوان بازاندیشی حریم شخصی با عنوان مابعد حریم شخصی که اسطوره های جهان واقعی در جهان رقمی را به چالش می کشد و طرح این پرسش که آیا در عصر این رایانش، حریم شخصی به معنای سنتی اعتباری دارد؟ پرسشی که در حوزه مشابهی نظیر حق تکثیر به مرزهای ناممکن یا بی اعتباری در جهان رقمی میل کرده است.

محتوای کتاب

کتاب دارای یک مقدمه، پیش گفتار، سه بخش، نه فصل و فهرست مراجع و منابع و یک نمایه است.

عناوین فصول کتاب به شرح زیر است:

- ۱- فصل یک: مقدمه: اسطوره و تنگنا در رایانش حی و حاضر
 - ۲- فصل دو: زمینه سازی رایانش حی و حاضر
 - ۳- فصل سه: جایابی برای ملاحظات اجتماعی و فرهنگی
 - ۴- فصل چهار: نقشی برای قوم نگاری: روشنگار و نظریه
 - ۵- فصل پنج: پشت پرده چه خبر
 - ۶- فصل شش: تحرک و شهرسازی
 - ۷- فصل هفت: بازاندیشی حریم شخصی
 - ۸- فصل هشت: زندگی خانگی و ملالت های آن
 - ۹- فصل نه: بازبینی رایانش حی و حاضر
- خواندن این کتاب ارزشمند و کمی دشوار را به مدرسان دانشگاهی رشته فا، پیگیران علوم بین رشته ای، پژوهشگران تبعات اجتماعی فناوری های نو و آینده نگاران فناوری می توان توصیه کرد.



عنوان کتاب: Architecture Principles

The Cornerstones of Enterprise Architecture

نویسندگان: Danny Greefhorst and Erik Proper

ناشر: Springer

زمان نشر: ۲۰۱۱

تعداد صفحات: 197 Pages

شابک: 978 - 3 - 642 - 20278 - 0

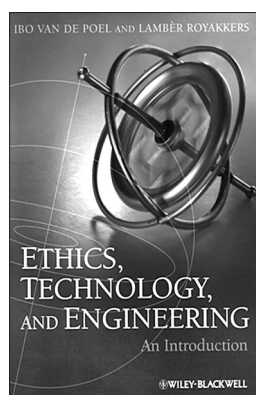
دارد که تمهیداتی برای کاهش این دشواری را در اولویت قرار می‌دهد که یکی از اولین راه‌حل‌ها استفاده از مدل‌های کارائی سنجی و ارسال نتایج مستمر اجرای آن‌ها از طریق پیشخوان‌های مدیریتی به مدیران است تا آن‌ها با اقدامات جبرانی یا پیشگیرانه و به موقع - در پی دریافت اطلاعات لحظه‌ای قابل استنتاج - از احتمال عدم توفیق پروژه‌ها در اجرا بکاهند. کتاب در زمینه معرفی و نمایش انواع پیشخوان‌های به ویژه معرفی امکانات رایانه در ارائه لحظه‌ای اطلاعات پایشی موفق است اما در وجه معرفی و نحوه معرفی مدل‌های کارائی سنجی بجز فهرست کردن معیارها توفیق چندانی ندارد. نویسندگان با عدم نگاه کاربردی به مفهوم معیار در مدیریت پروژه و عدم کمی‌سازی آن و نامگذاری کاربردی آن به آسا - ارزش‌های اندازه‌پذیر سازمانی^۱ - عملاً از پیوند منطقی آن با عوامل کلیدی توفیق^۲ و در پی آن نگاشت آن به شاخص‌های کلیدی کارائی (KPI) باز می‌ماند و به این ترتیب خواننده بدون اطلاع و فهم این نگاشت از کسب توانائی تولید معیارهای کارائی سنجی باز می‌ماند.

محتوای کتاب

کتاب دارای یک مقدمه، هشت فصل و یک نمایه است. عناوین فصول کتاب به شرح زیر است:

- ۱- فصل یک: تغییر چشم‌انداز در مدیریت پروژه.
- ۲- فصل دو: نیروهای محرکه برای معیارهای بهتر.
- ۳- فصل سه: معیارها.
- ۴- فصل چهار: شاخص‌های کلیدی کارائی.
- ۵- فصل پنج: معیارهای ارزش‌زای مدیریت پروژه.
- ۶- فصل شش: پیشخوان‌ها.
- ۷- فصل هفت: سامانه‌های کاربردی پیشخوانی
- ۸- فصل هشت: مدیریت پروژه اندازه‌پذیر

مطالعه این کتاب برای دانشجویان کارشناسی فضا، ضروری و برای مدیران پروژه اثربخش است.



عنوان کتاب: Ethics, Technology, and Engineering
An Introduction
نویسندگان: Ibo Van De Poel and Lamber Royakkers
ناشر: WILEY - BLACKWELL
زمان نشر: ۲۰۱۱
تعداد صفحات: 364 Pages
شابک: 978 - 1 - 4443 - 3094 - 6

1 - MOV: Measurable Organization Value

2 - CSF: Critical Success Factor

در حوزه رایانه و فناوری اطلاعات در شرایط فعلی است، باید شادمانه اظهار نماید که خوشبختانه برخی از این پایان‌نامه‌ها در لبه فناوری و با کیفیت قابل قبول قرار دارند از جمله پایان‌نامه‌ای که همین روزها در این حوزه در دانشکده مهندسی کامپیوتر دانشگاه صنعتی شریف با هدایت جناب آقای دکتر رامان رامسین توسط دانشجوی کارشناسی ارشد نرم‌افزار آقای محمد حسین احمدی در آستانه دفاع قرار دارد با عنوان "ایجاد مجموعه‌ای از الگوهای فرآیند برای معماری سازمانی"، که پس از مطالعه اجمالی آن - که به درخواست و لطف نگارنده آن صورت گرفت آشکارا گامی موفق در ادامه کار نویسندگان کتاب فوق را ترسیم می‌کند.

مطالب کتاب فوق برای همه دانشجویان، علاقه‌مندان و پیشه‌وران حرفه‌ای حوزه معماری سازمانی قابل توصیه است.

عنوان کتاب: Project Management : Metrics, KPIs, and Dashboards

A Guide to Measuring and Monitoring Project Performance

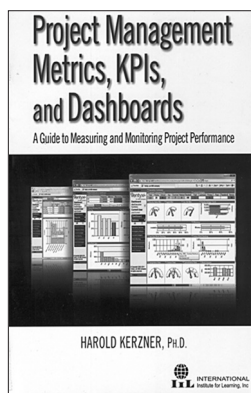
نویسنده: Harold Kerzner

ناشر: WILEY

زمان نشر: ۲۰۱۱

تعداد صفحات: 372 Pages

شابک: 978 - 1 - 118 - 02652 - 6



مقدمه

در پشت جلد کتاب محتوای آن در حوزه مدیریت معیارها در زمینه مدیریت پروژه طبقه‌بندی شده است و مطالعه آن را برای مدیران عملکردی واجد مزایای زیر دانسته است:

- یکپارچه‌سازی موفق شاخص‌های کلیدی کارائی (KPI: Key Performance Index) و معیارها در مدیریت یک پروژه با راهبردهای کسب و کاری.
- استفاده از مهمترین فنون پیشخوان (Dashboard) های کسب و کاری در نظارت بر کارائی پروژه‌ها.
- اشاره به مهمترین تفاوت‌های پیشخوان‌های متفاوت یک پروژه.
- مدیریت مقاومت در مقابل تغییرات.

در عنوان دوم، نویسندگان کتاب را راهنمایی برای اندازه‌گیری و نظارت بر کارائی پروژه‌ها نام گذاشته‌اند که محتوای کتاب موید آن است.

کتاب واجد یک نوآوری و بحثی مفصل در زمینه پیشخوان‌های مدیریتی در مدیریت کارائی پروژه و یک ایراد در نحوه معرفی مدل‌های مرجع کارائی سنجی است. نتایج ادواری پروژه CHAOS نشان از عدم توفیق حدود پنجاه درصدی پروژه‌های فا در جهان

مقدمه

لوئیس بوچارللی (Louis Bucciarelli) از موسسه فناوری ماساچوست در باره این کتاب می‌نویسد: نویسندگان کتابی جامع، روزآمد و متنی خواندنی نوشته‌اند. بحث‌های متنوع آن‌ها در زمینه‌های مختلف با چارچوب‌بندی، بسیاری از مشکلات اخلاقی که مهندسان با آن مواجه می‌شوند را پوشش می‌دهد. این موارد را آن‌ها از موضوعات معاصر استخراج و با مباحثه نحوه تصمیم‌گیری در موردشان را نشان می‌دهند. و این بهترین نحوه آموزش این مطالب در دوره کارشناسی است. کتاب پوئل و رویا کرز مقدمه‌ای بر آداب، فناوری و مهندسی نام دارد و بر پایه کتاب درسی آلمانی این دو و پی‌یترز در سال ۲۰۰۴ نوشته شده است. در مقدمه کتاب نویسندگان به تفاوت علوم و مهندسی از منظر خود اشاره کرده‌اند که مهندسان نه فقط برای فهم جهان بلکه برای تغییر آن تلاش می‌کنند. هر چند نگارنده اعتقاد دارد مهندسانی هم بوده اند که نفهمیده برای تغییر جهان تلاش کرده‌اند از مقصرین جهان غیرآدابی و مخرب محیط زیست (پر آسایش و کم آرامش) کنونی هستند. در همین مقدمه نویسندگان به زنجیره‌ای از مهارت‌های استدلالی اخلاقی در قالب یک مدل اشاره می‌کنند که بسیار مفید و کاربردی است. از این کتاب می‌توان به خوبی در کنار کتبی درسی در زمینه آداب فناوری اطلاعات نظیر کتاب رینولدز برای تدریس در دوره کارشناسی در درس آداب فا بهره گرفت.

محتوای کتاب

کتاب دارای یک پیشگفتار، مقدمه، ده فصل مطلب، پنج پیوست، واژه‌نامه، فهرست منابع، نمایه موارد مطالعاتی و یک نمایه است. عناوین فصول کتاب به شرح زیر است:

- ۱- فصل یک: مسئولیت‌های مهندسان.
- ۲- فصل دو: بندهای آدابی.
- ۳- فصل سه: آداب هنجاری.
- ۴- فصل چهار: استدلال‌های هنجاری.
- ۵- فصل پنج: چرخه آدابی.
- ۶- فصل شش: پرسش‌های آدابی در طراحی فناوری.
- ۷- فصل هفت: طراحی مدل اخلاقی.
- ۸- فصل هشت: جنبه‌های آدابی خطرات فنی.
- ۹- فصل نه: توزیع مسئولیت‌ها در مهندسی.
- ۱۰- فصل ده: پایداری، آداب و فناوری.

مطالعه این کتاب به همه مهندسان قابل توصیه است. کتابی که به اخلاق مهندسی از منظر مهندسی با دقت کافی نگاه کرده است.

اعضاء هیأت‌مدیره انجمن:

اعضاء هیأت مدیره:

آقای ابراهیم نقیب زاده مشایخ (رئیس)

آقای سید ابراهیم ابطحی (نایب رئیس)

آقای دکتر محمد صنعتی (دبیر)

آقای مهندس محمدحسن محوری (خزانه‌دار)

آقای دکتر علی فرخی

آقای دکتر علیرضا باقری

آقای دکتر عباس نوذری دالینی

آقای دکتر محسن صدیقی مشکنانی (عضو علی‌البدل)

آقای مهندس علیرضا ماهیار (عضو علی‌البدل)

کمیته‌های انجمن:

آقای ابراهیم نقیب‌زاده مشایخ (سرپرست کمیته

انتشارات)

آقای سید ابراهیم ابطحی (سرپرست کمیته

گردهمایی‌های علمی)

آقای مهندس محمدحسن محوری (سرپرست کمیته

عضویت و روابط عمومی)

آقای دکتر علی فرخی (سرپرست کمیته آموزش)



لیست اعضای حقوقی

انجمن انفورماتیک ایران

گسترش انفورماتیک ایران



نوید ایرانیان

(گسترش فضای مجازی)



مانیر

(مشاورین انفورماتیک نیرو)



مبنا

(مشاورین بهبود روش ها و سامانه های مبنا)



مجموع داده ها و سیستم ها (MDS)



مدار گسترش فناوری اطلاعات



مشاورین پگاه سیستم پیشرو



مهندسی نرم افزاری رایورز



نماد ایران



اعضای حقوقی فعال در حوزه نرم افزارهای کاربردی

آپادانا نگین پرداز



اطلاع رسانی پیوند داده ها



اندیشه پرداز ماهان



ایران رایانه



ایزایران



رایاوران توسعه



رای دانا آفرین



شرکت پایه ریزان راه کارهای

فراگیر



سامانه آرا پردازشگر



سبزه داده افزار



سحر رایانه



سند پرداز



سوایت پرداز



سیستم های اطلاعات مدیریت شرق رایا



شماران سیستم



طرح و پردازش فرا رایانه



عمید رایانه شریف (عرش)



فناوران انیاک



کرانه

(سامانه برنامه ریزی منابع کرانه)



کورش رایانه پیشرو



گروه نرم افزاری پیوست



اعضای حقوقی فعال در حوزه راه حل های برنامه ریزی منابع سازمان و نرم افزارهای پیشرفته سازمانی

آرا سامانه پویا



آراد رایانه لیان



ایران فاوا گسترش



ایریسا (بین المللی مهندسی سیستم ها و اتوماسیون)



برجاس (مهندسی مشاور برجاس)



بهکو (بهینه کوشان سپهر)



پارس رویال نفیس



پارس تصمیم



پردازش موازی سامان



تتا (تحقیق و توسعه ارتباطات)



تدوین فرآیند



توسعه فواصنعت



چارگون



داده پردازی دوران



داده پردازی نیلرام مانا



لیست اعضای حقوقی

انجمن انفورماتیک ایران

پدیسار انفورماتیک



پویا



تحلیلگران سیستم



توسعه سامانه‌های
نرم‌افزاری نگین (توسن)



خدمات انفورماتیک



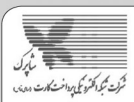
داده‌پردازی خوارزمی



داده‌پردازی ایران



شرکت شبکه الکترونیکی
پرداخت کارت شاپرک



فناوری اطلاعات و ارتباطات
پاسارگاد آریان (فناپ)



ماشین‌های تجاری بین‌المللی



صنایع یاس ارغوانی



اعضای حقوقی فعال در حوزه اینترنت
و پورتال‌ها

آزاد نت رسانه



ارتباطات مبین نت



مدیران سیستم پاسارگاد



مهندسی نرم‌افزار ایده‌گستر پوریا



موسسه تدبیرگران موج سوم



نوآوران مهر پارسه



همکاران سیستم



یکتاپردازان ویراسرای جنوب



اعضای حقوقی فعال در حوزه بانکی و
بانکداری الکترونیکی

اداره فناوری اطلاعات بانک سپه



الماس هوشمند ایرانیان



بانک اقتصاد نوین



بانک پارسیان



بانک سامان



بانک سرمایه



بانک ملی ایران



به‌پرداخت ملت



ایده پردازان اسطوره



ایده گستر کارین



آریانا پرداز آینه (آریا)



آیین رایانه اصفهان (آریکو)



داده پردازان احداث



داده‌پردازی حرفه‌ای‌ها



دی کاو ماندگار



رایانه خدمات امید



سارینا سیستم پرداز



سرو رایانه



سنجه حساب



شایگان سیستم



شرکت آلفا افزار



فرایپام



فرجام فرایپرداز



گاتا



لیست اعضای حقوقی

انجمن انفورماتیک ایران

تحلیلگران شبکه گستر پارسه



تدبیر فراز پویا (تفرا)



دانشگاه آزاد علوم تحقیقات
خوزستان



رایان مهر الکتریک



رهنمون فناوری اطلاعات



زاگرس سیستم ماهان



سامانه پرداز اریس



سیمرغ سامانه تهران



شارلوت رایانه



شبکه بانان پاژ



شبکه پایدار فناوری اطلاعات



شرکت مهندسی طرح و توسعه یاسین افزار



فن آوری اطلاعات و ارتباطات
آرمان تندیس ایرانیان (فن آرا)



فناوری نوین آپادانا



کلینیک هارد دیسک ایران (ihddc)



هاست ایران



اعضای حقوقی فعال در حوزه شبکه و سخت افزار

آتی نگر بابکان
(مجتمع انفورماتیک آتی نگر)



آداک فناوری مانییا



آروین رایان ارتباط قزوین



آریا همراه سامانه (سهامی خاص)



اروند رایان گستر



اندیشه نگار غرب



برد پرداز رایانه



بهینه سازان توان



پریس افزار رایانه



پرتو بیتا جاوید



پردیس فن آوری افق



پویا پرتو پیشرفته



تجارت هوشمند آوالان



ارتباطات شبکه‌های هوشمند
(تجارت الکترونیکی صبا)



افرانت



پردازشگر رایازما



پیشگامان دامنه فناوری



داده پرداز پویای شریف



روند تازه



ره پویان علم



سامانه ارتباطات
فلسک



فناوری اطلاعات و ارتباطات پارس
پردازش سدید



کیانا پارسیان کیش



کلیدگستر آینده (نت بینا)



گروه شرکت‌های فن آوا



مؤسسه گسترش اطلاعات و ارتباطات
و فرهنگی ندا رایانه



نخبگان تجارت



نوین پردازش زمرد



مهندسین مشاور پایابنیان صنعت



لیست اعضای حقوقی

انجمن انفورماتیک ایران

پژوهشکده فناوری اطلاعات
و ارتباطات



سازمان انرژی اتمی



سماتک



فناوری اطلاعات بین الملل



مرکز تحقیقات صنایع انفورماتیک



مؤسسه عالی مدیریت دانش



مؤسسه تحقیقات ارتباطات
و فناوری اطلاعات



مهندسی فناوری دانش نوید شرق



اعضای حقوقی فعال در حوزه مشاوره

توسعه شبکه هوشمند ارتباطات شفیق



فرامکان

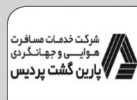


مشاوره مدیریت پیشنهاد سرویس



سایر حوزه‌ها

پارین گشت پردیس



آموزش و پرورش

آماج فناوری اطلاعات



و ارتباطات ایرانیا

خدمات مشاور خرد پیروز



دنیای مجازی کسب و کار



راهبر نیروی خراسان (رانیر)



زامیاد



سازمان بهداشت و درمان صنعت نفت



سایت رزرو آنلاین هتل
و راهنمای سفر زورق



فناوران سپاکو رایانه



کیسون



گروه کارخانجات چینی مقصود



گسترش فناوری اطلاعات
ستاره ویدا



مهندسی و ساخت بویلر مپنا



مهندسی همکارییمان البرز



مرکز گسترش فناوری اطلاعات (مگفا)



اعضای حقوقی فعال در حوزه آموزش و پژوهش

آموزشکده فنی و حرفه‌ای سما
واحد کرج- برادران



گروه مهندسين فرايند



مهندسی بهین پژوهش خاور



مهندسی پارتیان ابتکار پایدار



مهندسی رایان توان افزار



میزان گستر رایانه



یکتا گامان پیشرو



اعضای حقوقی فعال در حوزه مدیریت پروژه

شرکت فناوری اطلاعات سبحان رایان مبارکه
(فاسکو)



فناوران اطلاعات بهاران



اعضای حقوقی مشتریان بهره‌بردار از راه‌حل‌های نرم‌افزارهای پیشرفته سازمانی و بهره‌بردار از فناوری اطلاعات

انرشمی



ارتباط پردازان تجارت ایرسا



پارسین تجارت پایا کیش



پردازش هوشمند البرز



خدمات مسافرتی شهرزاد



گرید: تور، تورین، تورینه

سیامک کاظمی

واژه «شبکه» چون قبلاً برای net, network و حتی lattice به کار رفته در این مورد قابل استفاده نیست. «مشبکه» هم خیلی قدمایی و سنگین است. می‌ماند لغت‌های خانواده «تور» (یا لغت‌های دیگری که صاحب‌نظران پیشنهاد کنند). از این خانواده، واژه «تورین» شانس آن را داشته که در پژوهشگاه دانش‌های بنیادی، دست کم در نوشتار، کمابیش رایج شود. اما «تورین» صفت منسوب به اسم است نه اسم. اگر هم، با قدری تسامح، آن را جانشین اسم بگیریم، باید grid را در حالت صفتی، «تورینی» بگوییم مثلاً «رایانش تورینی» برای grid computing. واژه «تورینه» هم سنگین به نظر می‌رسد. شاید بهتر باشد ساده‌ترین عضو این خانواده یعنی «تور» را به کار ببریم و نگران نباشیم که مثلاً با تور ماهیگیری اشتباه می‌شود یا از ابهت و قدر و منزلت grid می‌کاهد. در صورت انتخاب «تور»، می‌توان grid computing را «رایانش تورین» و computational grid را «تور محاسباتی یا رایانشی» گفت.

کاربرد و اهمیت روزافزون grid در محیط‌های علمی ایجاب می‌کند واژه مناسب واحدی برای آن در زبان فارسی داشته باشیم. امیدواریم با مشارکت اهل فن در این بحث و مساعدت فرهنگستان زبان فارسی، چنین واژه‌ای انتخاب و تثبیت شود. به نقل از مجله اخبار، پژوهشگاه دانش‌های بنیادی، شماره پیاپی ۶۲، ۱۳۹۱.

اصطلاح کامپیوتری grid را در فارسی با چه واژه‌ای بیان کنیم؟ آیا باید عین همین کلمه تازه‌وارد را (به صورت «گرید» در نوشته‌ها) به کار برد یا برابر نهاده‌ای فارسی (تور، توری، تورینه، تورین، شبکه، مشبکه، یا چیز دیگری) به جای آن برگزید؟ به نظر نویسنده این سطور (و عده‌ای از متخصصان و دست‌اندرکاران کامپیوتر) کاربرد این واژه انگلیسی در فارسی چندان پسندیده نیست بخصوص که صورت نوشتاری آن ممکن است با گرید (grade) که لغت آشناتری است، اشتباه شود و صفت منسوب به آن را هم باید «گریدی» گفت که مقبول به نظر نمی‌رسد. وانگهی این کلمه در زبان معمولی انگلیسی معنای تخصصی عمیقی ندارد که هیچ واژه فارسی از عهده بیان آن برنیايد.

واژه grid در زبان انگلیسی یک کلمه کاملاً معمولی است که معنی اصلی‌اش شبکه یا توری است و انگلیسی زبان غیرمتخصص هم همین معنای کلی را از آن دریافت می‌کند. به عبارت دیگر، اهل کامپیوتر یک کلمه معمولی زبان را برگزیده و مفهوم تخصصی مورد نظر خود را بر آن «بار کرده‌اند» و این شیوه - بار کردن معانی جدید تخصصی بر لغات معمولی و عمومی - یک شیوه رایج واژه‌گزینی در زبان انگلیسی است. چرا ما در فارسی این کار را نکنیم؟

اما از میان معادل‌های فارسی که در بالا برای grid ذکر شد،

رتبه‌بندی‌ها و ویژگی‌های دانشگاه‌های برتر جهان*

حمیدرضا آراسته

دانشیار دانشگاه تربیت معلم، دانشکده روانشناسی و علوم تربیتی، مدیریت آموزش عالی، تهران، ایران.

پست الکترونیکی: arasteh@tmu.ac.ir

سید طاهر فاضلی ماسوله

دانشجوی دکتری مدیریت آموزش عالی، دانشگاه آزاد اسلامی، واحد علوم و تحقیقات، تهران، ایران

پست الکترونیکی: st.fazeli@srbiau.ac.ir

چکیده:

طی سال‌های اخیر ارزیابی و رتبه‌بندی دانشگاه‌های جهان توجه محققان و مسئولان آموزش عالی کشورها را به خود جلب کرده است. یکی از نتایج حاصل از این رتبه‌بندی‌ها ظهور مفهوم دانشگاه‌های برتر جهانی است. این نوع دانشگاه‌ها که در صدر فهرست رتبه‌بندی‌های جهانی قرار دارند، با ویژگی‌های خاص خود دارای بهترین بازدهی علمی و پژوهشی هستند. به همین دلیل، بسیاری از دانشگاه‌ها با الگو قرار دادن این قبیل دانشگاه‌ها در صدد نیل به جایگاه و نیز ویژگی‌های این قبیل دانشگاه‌ها هستند. در این مقاله با بررسی ابعاد مفهومی دانشگاه‌های برتر جهان، روش‌شناسی رتبه‌بندی‌ها و ویژگی‌های این گونه دانشگاه‌ها تبیین شده است.

واژه‌های کلیدی:

دانشگاه‌های برتر جهان، رتبه‌بندی، مؤسسات بین‌المللی رتبه‌بندی دانشگاه‌ها، تمرکز بر استعداد، وفور منابع، مدیریت مطلوب.

۱. مقدمه

دانشگاه‌ها منبع الهام‌بخش و عامل اصلی رشد و پیشرفت هر جامعه‌ای محسوب می‌شوند. این سازمان‌های اجتماعی، در عصری که پایه و بنیان آن بر خردورزی و جهل‌ستیزی قرار دارد، نقش یگانه‌ای در تحقق آرمان‌های بشری همچون حاکمیت عقل‌گرایی، صلح و عدالت و اخلاق داشته‌اند. بدون شک، جوامعی که به گسترش آزادی علمی و

استقلال دانشگاهی در محیط دانشگاه‌های خود توجه داشته و ضمن پاس داشتن آن به مقوله پیشرفت و توسعه دانشگاه اهتمام ورزیده‌اند، از موهبتات و تأثیرات مثبت آن در عرصه‌های مختلف سیاسی، اقتصادی، اجتماعی و فرهنگی بیشتر بهره‌مند بوده‌اند.

امروزه، دانشگاه‌ها با توجه به رسالت خود در خصوص رفع مشکلات و نیازمندی‌های ملی و فراملی تلاش‌های زیادی را به منظور پاسخگویی به نیازهای ذینفعان در دستور کار خود قرار داده‌اند و در این مسیر رقابت‌های ملی و بین‌المللی بین آن‌ها افزایش یافته است. رقابت میان دانشگاه‌ها در عرصه ملی و بین‌المللی سبب شده است که متولیان و ذینفعان دانشگاه‌ها با شناخت هر چه بیشتر نتایج سودمند این رقابت، به زمینه‌سازی و کمک به دانشگاه‌های متبوع خود برای کسب موفقیت بیشتر در امر رقابت با سایر دانشگاه‌ها بپردازند. بر اساس پیمایش هیزل کورن^۱، که با حمایت سازمان توسعه و همکاری‌های اقتصادی^۲ و انجمن بین‌المللی دانشگاه‌ها^۳ در سال ۲۰۰۶ صورت گرفت، غالب رؤسای دانشگاه‌ها تلاش‌هایی را برای ارتقای دانشگاهشان انجام داده‌اند. برای مثال، کشورهایی چون چین، فنلاند، فرانسه، آلمان، هند، ژاپن، لتونی، مالزی، روسیه، سنگاپور، اسپانیا، کره جنوبی، تایوان و ویتنام سیاست‌هایی را با هدف ایجاد دانشگاه‌های برتر تدوین و اجرا کرده‌اند. به‌طور کلی، این سیاست‌ها شامل تأمین بودجه بیشتر برای دانشگاه‌های

1- Hazelkorn

2- OECD

3- IAU

* این مقاله از فصلنامه آموزش مهندسی ایران شماره ۵۲، سال سیزدهم، زمستان ۱۳۹۰ نقل گردیده است.

۳. یافته‌های مطالعه

اگر چه موضوع کیفیت و ارتقای آن در عرصه فعالیت‌های آموزشی، تحقیقاتی و خدماتی در دانشگاه‌ها سابقه زیادی در تحقیقات و پژوهش‌های محققان در این زمینه دارد، واژه دانشگاه برتر جهان طی ده سال اخیر در میان اصحاب تحقیقات عرصه آموزش عالی و صاحب‌نظران دانشگاه‌ها مطرح شده است. به‌طور کلی، طی دهه اخیر زنجیره‌ای از تحولات به ویژه در عرصه اقتصاد و فناوری زمینه‌های بروز و ظهور اصطلاح دانشگاه برتر جهانی را در سپهر آموزش عالی سبب شده است. در این میان، مسئله اقتصاد دانش بنیان مهم‌ترین زمینه نضج و گسترش رقابت میان دانشگاه‌ها را فراهم آورده است.

از جمله پیشگامان بررسی دانشگاه‌های برتر جهان فیلیپ آلتباخ^۵ است که در مقاله‌ای با عنوان «هزینه‌ها و منافع دانشگاه‌های برتر جهانی: یک رهیافت آمریکایی» ابعاد گوناگون این نوع دانشگاه‌ها را شرح داده است [۳]. همچنین، کیائو دنگ و همکاران^۶ در مقاله‌ای با عنوان «ابتکارات ملی در تأسیس دانشگاه‌های برتر جهان: مقایسه تجارب کشورهای آسیایی و اروپایی» با بررسی مفاهیم نظری مرتبط با شکل‌گیری دانشگاه‌های برتر جهان، مهمترین اقدامات برخی از کشورهای آسیایی و اروپایی را در قالب برنامه‌های راهبردی در راه‌اندازی یا تبدیل دانشگاه‌ها به برترین رتبه‌های جدول دانشگاه‌های برتر بیان کرده‌اند. البته، به دلیل پیوند وثیق موضوع دانشگاه‌های برتر جهان با مسئله رتبه‌بندی دانشگاه‌ها در این خصوص پژوهش‌های گسترده دیگری صورت گرفته است [۴].

جان سادلر و لیو لیان کای (۲۰۰۹) در مقاله‌ای با عنوان «جایگاه جهانی: آرمان‌ها و آزمون واقعیت» عنوان می‌کنند که تحقق دانشگاه برتر جهان مستلزم فراهم کردن شرایط و امکاناتی است که بدون آن‌ها برخورداری از آن رؤیایی بیش نیست. نویسندگان در این مقاله ضمن تشریح استدلال خود در خصوص تلاش کشورها برای برخورداری از این نوع دانشگاه، شرایطی را که از آن به عنوان «شرایط منازعه آمیز در رقابت بین کشورها» یاد می‌شود، بیان می‌کنند. سادلر در مقاله خود توجه به دروندادهای دانشگاهی به ویژه برخورداری از زنجیره نوآوری و خلاقیت در امر پژوهش و سرمایه‌گذاری بر روی پژوهشگران را مورد تأکید قرار می‌دهد [۱۲].

لیو و چنگ (۲۰۰۵) در مقاله خود با عنوان «رتبه‌بندی دانشگاه‌های جهان، روش‌شناسی و مشکلات» به تشریح چگونگی بررسی مؤسسات رتبه‌بندی از عملکرد دانشگاه‌های سراسر دنیا می‌پردازند و ضمن تبیین روش‌شناسی مؤسسات رتبه‌بندی کننده دانشگاه‌ها، پارادوکس‌ها و مشکلاتی را که در این روش‌شناسی‌ها رخ می‌دهد، بیان می‌کنند. نویسندگان مقاله با بیان اشکالات روش‌شناسی مؤسسات رتبه‌بندی

برتر کشورهایشان و ادغام دانشگاه‌های کوچک و مؤسسات تحقیقاتی با دانشگاه‌هاست. حتی در برخی ایالات در آمریکا (برای مثال، ایالت آریزونا و فلوریدا) حقوق و مزایای رؤسای دانشگاه‌ها بر اساس جایگاه دانشگاه در رتبه‌بندی دانشگاه‌ها در سطح ملی تعیین می‌شود [۶].

موضوع ارزیابی و اعتبارسنجی فعالیت‌های دانشگاه‌ها در سطح محلی و ملی در گذشته به منظور کسب رتبه‌های مناسب در رقابت‌های دانشگاهی در کشورها سابقه بسیاری دارد. برای مثال، در سال ۲۰۰۳ فقط در کشور آمریکا هشت سازمان اعتبارسنجی منطقه‌ای، یازده انجمن یا شورای ملی اعتبارسنجی تخصصی و بیش از ۶۰ سازمان تخصصی و حرفه‌ای فعال بوده‌اند [۲۱]. اما مقوله پایش و ارزیابی فعالیت‌های دانشگاه‌ها در سطح بین‌المللی موضوعی جدید و نوظهور است که به ویژه در پایان قرن بیستم و ابتدای هزاره جدید بسیار مورد توجه نهادهای آموزش عالی قرار گرفته است.

در نتیجه استمرار رتبه‌بندی دانشگاه‌ها در سال‌های اخیر، مفهوم دانشگاه برتر جهان به عنوان یک اصطلاح مطلوب به منظور گسترش ظرفیت برای رقابت کردن در بازار آموزش عالی جهانی از طریق فراگیری، انطباق و تولید دانش پیشرفته به کار رفته است. اما عضویت در گروه به شدت رقابتی دانشگاه‌های برتر جهان به آسانی صورت نمی‌پذیرد، بلکه تحقق این امر نتیجه برنامه‌ریزی دقیق، مداومت در پیگیری اهداف در عرصه‌های آموزشی، پژوهشی و فناوری‌های کاربردی با مشارکت علاقه‌مندان و مجدانه اعضای هیأت علمی، دانشجویان و مدیریت دانشگاه است.

در این مقاله پس از مروری کوتاه بر مفهوم دانشگاه‌های برتر جهان، پیشینه تحقیقات در زمینه این گونه دانشگاه‌ها بررسی و سپس، به رتبه‌بندی دانشگاه‌های جهان در سال ۲۰۱۰، روش‌شناسی رتبه‌بندی مؤسسات بین‌المللی و در نهایت، ویژگی‌های دانشگاه‌های برتر جهان اشاره شده است.

۲. روش مطالعه

نظر به ماهیت تحقیق استفاده از ابزارهایی از قبیل مصاحبه، پرسشنامه، مشاهده و نظایر آن از سوی پژوهشگران سودمند ارزیابی نشد. اما تحلیل محتوا به عنوان روشی کیفی و نظام‌مند به منظور خلاصه کردن اسناد در دسته‌های مختلف بر اساس دستورالعمل‌های کدگذاری روشی مفید برای این مطالعه تشخیص داده شد [۱۹]. برای انجام یافتن این مطالعه ابتدا پژوهش‌ها، گزارش‌های رسمی و رسانه‌ای و اخبار روزنامه‌ها در بانک‌های اطلاعاتی و وبگاه‌های اینترنتی جستجو و سپس، اطلاعات بر اساس رهنمودهای استرن لر^۴ دسته‌بندی شد [۱۴]. استفاده از این روش امکان بررسی نظام‌مند اسناد را در چهار دسته مفاهیم رتبه‌بندی جهانی، روش‌شناسی رتبه‌بندی در مؤسسات بین‌المللی، عوامل تقویت‌کننده و ویژگی‌های دانشگاه‌های برتر جهان برای پژوهشگران فراهم کرد.

5- Philip Altbach

6- Qiao Deng et al.

4- Sternler

در دانشگاه‌ها استفاده می‌شود، به طور کلی در همه شاخص‌ها بر کیفیت آموزش، بین‌المللی شدن، محصولات پژوهشی و اثرگذاری تأکید زیادی می‌شود [۱۳]. لیو (۲۰۰۹) نیز تعریفی کاربردی از دانشگاه‌های برتر جهان ارائه کرده است. وی معتقد است که دانشگاه برتر جهان نهادی است که آموزش و تحقیقات آن مبتنی بر استانداردهای بین‌المللی در طیفی گسترده از رشته‌هاست و تلاش‌های فراوانی را به منظور خدمت‌رسانی در دو سطح ملی و بین‌المللی انجام می‌دهد [۸].

بر اساس تعاریف مذکور، سالمی مجموعه‌ای از ویژگی‌های دانشگاه‌های برتر جهان را شامل تأکید زیاد بر استعداد [اعم از استادان و دانشجویان]، فراوانی منابع [برای ارائه آموزش غنی و فراهم کردن محیط پژوهش] و اداره کردن خودگردان و مشتاقانه دانشگاه [برای تشویق و ترغیب خلاقیت و داشتن بینش راهبردی] معرفی می‌کند.

۵. رتبه‌بندی دانشگاه‌ها در سال ۲۰۱۰

دو مؤسسه رتبه‌بندی دانشگاه‌ها (مؤسسه تایمز و شانگ‌های) برای سال ۲۰۱۰ فهرست بهترین دانشگاه‌های برتر جهان را ارائه کردند. مؤسسه تایمز در روش‌شناسی خود از عوامل ذهنی^{۱۲} و عینی^{۱۳} و نیز دیدگاه‌های صاحب‌نظران با توجه به رتبه‌های سال قبل دانشگاه‌ها استفاده می‌کند. در گزارش اخیر این مؤسسه که همانند گذشته در برگیرنده ۲۰۰ دانشگاه دنیاست، در میان ۲۰ دانشگاه برتر جهان، ۱۵ دانشگاه در کشور آمریکا، سه دانشگاه در کشور انگلستان، یک دانشگاه در کشور سوئیس و یک دانشگاه در کشور کانادا قرار دارد. همانند سال‌های گذشته، دانشگاه هاروارد در صدر این گروه با امتیاز ۹۶/۱ قرار گرفته است و در رتبه بیستم دانشگاه کارنگی ملون آمریکا با ۷۳/۹ امتیاز قرار دارد [۱۷].

از سوی دیگر مؤسسه رتبه‌بندی دانشگاه‌های جهان متعلق به دانشگاه شانگ‌های نیز که سالیانه فهرست ۵۰۰ دانشگاه دنیا را منتشر می‌کند، با لحاظ کردن شاخص‌هایی همچون تعداد برندگان جایزه نوبل و فیلدز و مقالات نمایه شده در مجلات معتبر، فهرست ۵۰۰ دانشگاه دنیا در سال ۲۰۱۰ را منتشر کرده است. در این رتبه‌بندی به هر شاخص حداکثر ۱۰۰ نمره داده می‌شود که در نهایت با تحلیل فعالیت‌های هر دانشگاه مرتبه آن در رتبه‌بندی جهانی تعیین می‌شود. در گزارش اخیر این مؤسسه همچون فهرست مؤسسه تایمز در میان بیست دانشگاه برتر جهانی دانشگاه هاروارد رتبه نخست را به خود اختصاص داده است. در میان این ۲۰ دانشگاه، ۱۷ دانشگاه آمریکایی و ۲ دانشگاه انگلیسی و یک دانشگاه ژاپنی حضور دارند [۱].

دو مؤسسه رتبه‌بندی تایمز و شانگ‌های علاوه بر ارائه فهرست‌های کلی از رتبه‌های دانشگاه‌های برتر، به رتبه‌بندی دانشگاه‌ها در

کننده دانشگاه‌ها، از حاکمیت نتایج نادرست اعلام شده آنان در سطح جهانی انتقاد می‌کنند [۹].

همچنین، وحید رضا اوحدی (۱۳۸۶) در مقاله‌ای با عنوان «معیارهای رتبه‌بندی دانشگاه‌ها» با تمرکز بر بررسی جزئیات روش‌شناسی سه مؤسسه بین‌المللی رتبه‌بندی دانشگاه‌ها یعنی تایمز، دانشگاه شانگ‌های و مؤسسه سایبرمتریک، ویژگی‌های مختص هر یک را تشریح می‌کند. وی تأکید می‌کند که احراز رتبه‌های برتر در رتبه‌بندی‌های جهان با پذیرش دانشجویان با توانایی بالاتر و در نتیجه، بهبود کیفیت بدنه دانشجویی دانشگاه رابطه مستقیم دارد. وی هدف از مقاله خود را مقایسه سه روش بین‌المللی رتبه‌بندی دانشگاه‌ها و تعیین جایگاه دانشگاه‌های جهان اسلام در این روش‌های رتبه‌بندی عنوان می‌کند [۲۰].

از مفصل‌ترین و جدی‌ترین فعالیت‌های پژوهشی که در سال‌های اخیر در خصوص دانشگاه‌های برتر جهان صورت گرفته است، می‌توان به پژوهش جمیل سالمی، مشاور عالی رئیس بانک جهانی، اشاره کرد [۱۳]. سالمی در کتاب خود با عنوان «چالش ایجاد دانشگاه‌های برتر جهان»، با مطالعه ویژگی‌های دانشگاه‌های برتر و نیز روش‌شناسی مؤسسات رتبه‌بندی جهانی، ضمن بررسی مفهوم دانشگاه برتر جهان، الگویی جامع از ویژگی‌های این نوع دانشگاه ارائه کرده است. سالمی در تحقیق خود سه عامل کلیدی را در ایجاد دانشگاه‌های برتر بر می‌شمرد که عبارت‌اند از: تمرکز استعداد^۷، کثرت منابع^۸ (حمایتی) و اداره کردن مطلوب^۹.

۴. مفهوم دانشگاه برتر جهانی

علی‌رغم کثرت استفاده از واژه دانشگاه‌های برتر جهان در عرصه نظام آموزش عالی، توافق عامی در این باره وجود ندارد. به زعم آلتباخ ارائه نشدن تعریفی روشن باعث شده است که «هر کسی به دنبال دستیابی به دانشگاه برتر جهان باشد، اما هیچ کس نمی‌داند که این پدیده چیست و هیچ کس هم مشخص نکرده است که چگونه دست یافتنی است». آلتباخ دانشگاه‌های برتر جهان را چنین تعریف می‌کند: دانشگاه‌های برتر جهان دانشگاه‌هایی هستند که از عالی‌ترین سطوح اعضای هیأت علمی، ممتازترین پژوهش‌ها، کیفیت برتر آموزشی و مبتنی بر استانداردهای بین‌المللی، منابع دولتی و غیردولتی، دانشجویان بسیار مستعد، آزادی علمی، ساختارهای مدیریتی مستقل و امکانات پیشرفته برای آموزش، پژوهش، مدیریت و زندگی دانشجویی برخوردارند [۲].

برخی از صاحب‌نظران تلاش کرده‌اند تا از طریق جداول امتیازدهی دانشگاه‌ها مانند آنچه به وسیله مؤسسه تایمز^{۱۰} و دانشگاه شانگ‌های جیائو تانگ^{۱۱} منتشر می‌شود، این تعریف را ارائه دهند. با وجود روش‌های متفاوتی که در ارزیابی‌های بین‌المللی رتبه‌بندی

7- Concentration of Talent

8- Abundant Resources

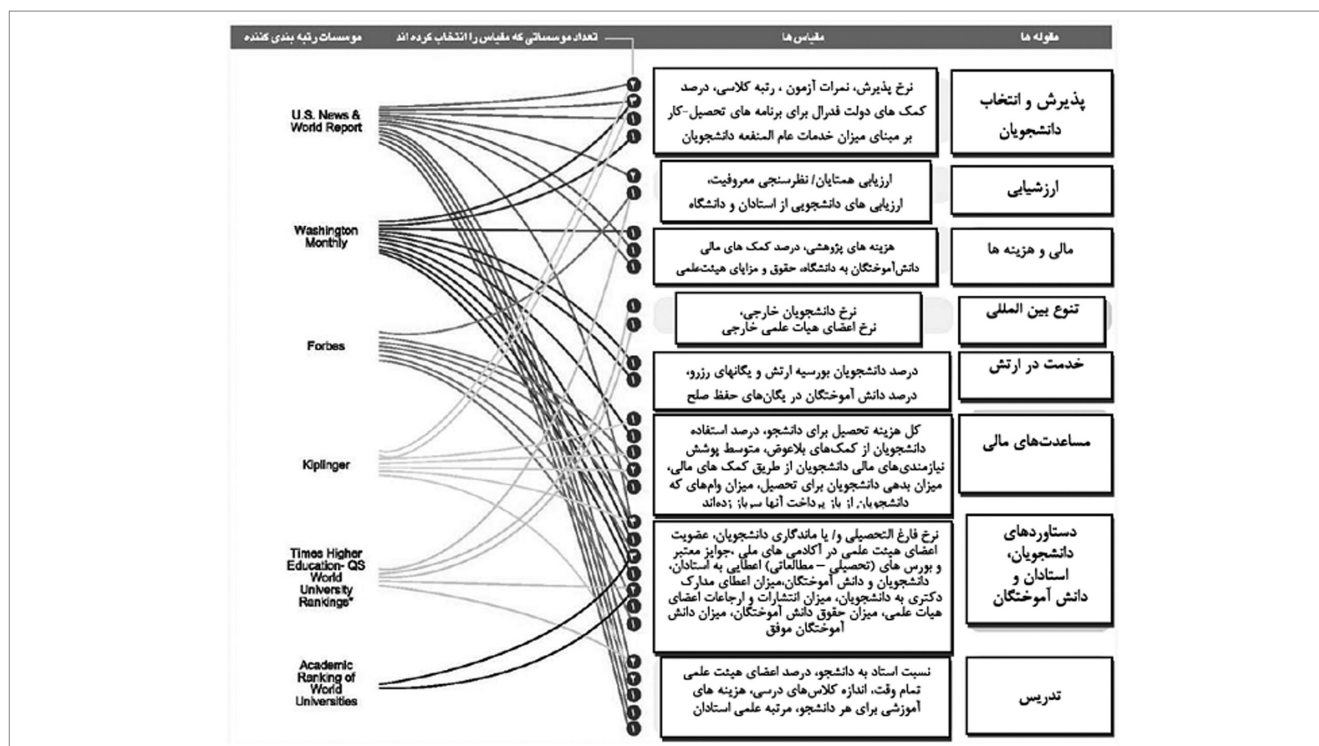
9- Favorable Governance

10- The Times Higher Education Supplement

11- Shanghai Jiao Tong University

12- Subjective

13- Objective



شکل ۱: شاخص های شش مؤسسه بین المللی برای رتبه بندی

بازمی گردد، به گونه ای که به واسطه سیاست آموزش مهندسی در کشور ژاپن، ۲۱/۶ درصد از کل دانشجویان این کشور در رشته های مهندسی تحصیل می کنند. این درصد در دنیا همانند ندارد، به گونه ای که در کشور آمریکا ۵/۹ درصد، در کشور انگلستان ۱۶/۷ درصد و اروپای غربی ۱۵ درصد از کل دانشجویان در رشته های مهندسی تحصیل می کنند [۲۲].

۶. روش شناسی رتبه بندی در مؤسسات بین المللی

طی سالیان اخیر شاهد ظهور مؤسسات فعال در زمینه ارزیابی فعالیت های علمی پژوهشی دانشگاه ها و نیز رتبه بندی آن ها بر اساس استانداردها و شاخص های خاص بوده ایم که این امر بر مقوله رقابت بین دانشگاهی در عرصه بین الملل تأثیر بسزایی داشته است. مؤسسات بین المللی رتبه بندی دانشگاه ها شاخص های گوناگونی را مد نظر قرار می دهند و هر یک از این نظام ها روش های خاص و اهداف متفاوتی را دنبال می کنند. ریچارد و کادینگتون^{۱۴} شاخص های شش مؤسسه بین المللی را در هشت دسته و میزان کثرت ارجاع به هر یک از شاخص ها را به شرح شکل ۱ جمع بندی کرده اند [۱۱]. همان گونه که مشاهده می شود، برخی از شاخص ها بیشتر مورد توجه مؤسسات شش گانه قرار گرفته است. برای مثال، نرخ دانش آموختگی یا ماندگاری دانشجویان با امتیاز ۴ بیشترین استفاده را در میان مؤسسات رتبه بندی دارد. همچنین، مقیاس های نمرات

رشته ها و حوزه های مطالعاتی نیز می پردازند. برای مثال، مؤسسه تایمز بر اساس موضوعات مهندسی و فناوری، هنر و علوم انسانی، علوم پزشکی، علوم طبیعی، علوم فیزیکی و علوم اجتماعی رتبه بندی دانشگاه ها را انجام می دهد. در رتبه بندی سال ۲۰۱۱ مؤسسه تایمز، در صدر برترین دانشگاه های مهندسی جهان، دو مؤسسه تکنولوژی کالیفرنیا و مؤسسه تکنولوژی ماساچوست آمریکا توأمان مرتبه اول را به خود اختصاص داده اند. در ادامه این فهرست به ترتیب تا رتبه دهم، دانشگاه های پرینستون، کالیفرنیا - برکلی، استنفورد، کمبریج، آکسفورد، کالیفرنیا - لس آنجلس، مؤسسه تکنولوژی زوریخ سوئیس و امپریال کالج لندن قرار گرفته اند [۱۶].

مؤسسه شانگهای چین نیز در رتبه بندی برترین دانشگاه های مهندسی جهان همانند تایمز، مؤسسه تکنولوژی ماساچوست را در صدر جدول و در ادامه دانشگاه های استنفورد، کالیفرنیا - برکلی، ایلینویز، مؤسسه تکنولوژی جورجیا، دانشگاه تگزاس، دانشگاه میشیگان، دانشگاه کارنگی ملون، دانشگاه پنسیلوانیا و دانشگاه کالیفرنیا در سان دیگو را قرار داده است.

نکته قابل توجه در آمار ارائه شده جایگاه ویژه دانشگاه های آسیایی به ویژه کشور ژاپن در فهرست برترین دانشگاه های مهندسی دنیاست. در این خصوص دو دانشگاه توکیو با رتبه ۲۵ و کیوتو با رتبه ۳۲ جزو برترین دانشگاه ها به حساب می آیند. یکی از مهمترین دلایل رشد جایگاه دانشگاه های کشور ژاپن به سرمایه گذاری وسیع این کشور در حوزه آموزش مهندسی

14- Richards and Coddington

می‌کنند. برای مثال، دانشگاه پکن همه ساله ۵۰ نفر از بهترین دانشجویان هر استان این کشور را جذب می‌کند. همچنین، دانشگاه هاروارد، دانشگاه تکنولوژی کالیفرنیا، دانشگاه تکنولوژی ماساچوست و دانشگاه ییل، گزینشی‌ترین دانشگاه‌ها در ایالات متحده آمریکا هستند. از سوی دیگر، دانشگاه‌های برتر جهان سعی در حفظ تناسب دانشجویان دوره‌های تکمیلی نسبت به دوره‌های پایین‌تر به منظور حفظ تأثیر فعالیت‌های پژوهشی دانشجویان خود دارند.

بین‌المللی‌شدن دانشگاه‌های برتر همچنین، موجب ایجاد دانشگاه‌هایی متشکل از نخبگان علمی شده است. هر دو مؤسسه بین‌المللی رتبه‌بندی دانشگاه‌های جهان یعنی تایمز و شانگ‌های به مقوله بین‌المللی بودن دانشگاه‌های برتر توجه ویژه‌ای دارند. در بیشتر موارد، دانشگاه‌های برتر جهان بر خوردار از دانشجویان و استادانی هستند که در سایر دانشگاه‌ها فعال بوده‌اند. دانشگاه‌های برتر جهان با ایجاد جذابیت برای دانشجویان و استادان ممتاز و پذیرش ایده‌های جدید و رهیافت‌های نو آنان را جذب خود می‌کنند.

بهره‌مندی از منابع فراوان (به منظور فراهم ساختن یک محیط مستعد یادگیری و انجام دادن پژوهش‌های پیشرفته)

وفور منابع دومین عاملی است که دانشگاه‌های برتر جهانی را قادر می‌سازد تا هزینه‌های ناشی از فعالیت‌های علمی و پژوهشی گسترده و شدید را تأمین کنند. به طور کلی، دانشگاه‌های برتر جهان از چهار منبع مالی حمایت‌کننده زیر برخوردار هستند:

- الف. بودجه‌های دولتی ویژه فعالیت‌های علمی و پژوهشی؛
- ب. قراردادهای پژوهشی با بخش‌های دولتی و خصوصی؛
- پ. درآمدهای ناشی از موقوفات و هدایا؛
- ت. شهریه‌های دانشجویی.

در کشورهای اروپای غربی بودجه‌های دولتی مهم‌ترین منبع آموزش و پژوهش است، هر چند دانشگاه‌های برتر انگلستان از برخی منابع وقفی و شهریه‌های بالا که در سال‌های اخیر مطرح شده است نیز برخوردارند. در آسیا دانشگاه ملی سنگاپور که در سال ۲۰۰۶ به یک دانشگاه خصوصی تبدیل شد، موفق‌ترین مؤسسه در جذب منابع وقفی قابل توجه بوده است. این دانشگاه موفق شده است با یک سهام چشمگیر ۷۷۴ میلیون دلاری ثروتمندتر از دانشگاه‌های کمبریج و آکسفورد انگلستان شود.

در کشور آمریکا توان مالی دانشگاه‌های برتر ناشی از دو عامل است: نخست آن که آن‌ها از میزان گسترده‌ای از موقوفات بهره‌مندند که این موضوع باعث می‌شود تا آن‌ها آسوده‌تر بر اولویت‌های میان مدت و بلندمدت خود تمرکز داشته باشند. برای مثال، موقوفات دانشگاه هاروارد در سال ۲۰۱۰ بیش از ۲۷/۵ میلیارد دلار و بودجه عملیاتی آن ۳/۷۲۵ میلیارد دلار بود [۵]. دوم، میزان بهره‌مندی آنان از اعضای

آزمون، جوایز معتبر و بورس‌های اعطایی به دانشجویان، استادان و دانش‌آموختگان با امتیاز ۳ دارای بیشترین استناد توسط مؤسسات یاد شده بودند. در این میان برخی از شاخص‌ها همچون رتبه کلاسی یا هزینه‌های پژوهشی کمترین استناد را در میان مؤسسات رتبه‌بندی بین‌المللی دانشگاه‌ها به خود اختصاص داده‌اند. شایان ذکر است که برخی از مقیاس‌ها بومی شده برای کشوری خاص است. برای مثال، تعداد دانش‌آموختگان شرکت‌کننده در یگان پاسدار صلح یا درصد دانشجویان بورسیه ارتش و یگان‌های رزرو صرفاً برای کشور آمریکاست.

۷. ویژگی‌های دانشگاه‌های برتر جهان

همان‌گونه که پیشتر بدان اشاره شد، برخی از صاحب‌نظران در ارائه تعریف از دانشگاه‌های برتر جهان به بیان برخی ویژگی‌های آن‌ها پرداخته و از این طریق وجوه ممیزه و تفکیک‌کننده این نوع دانشگاه‌ها را از سایر دانشگاه‌ها بیان کرده‌اند. اگرچه در نظر بسیاری خصوصیات عام همچون برخورداری از شهرت جهانی، منابع وسیع حمایتی و بهره‌مندی از اعضای هیأت علمی ممتاز به عنوان نخستین ویژگی‌های این قبیل دانشگاه‌ها محسوب می‌شود، باید خاطر نشان ساخت مادامی که با یک چارچوب نظری و ذهنی به مقوله‌بندی عوامل شکل‌دهنده این نوع دانشگاه اقدام نشود، دائماً دچار نقص در بیان ویژگی‌های آن خواهیم شد. بر این اساس، به نظر می‌رسد که طراحی یک الگوی نظری منطبق بر واقعیات دانشگاه‌های برتر جهان موجب شناخت همه جانبه‌تر و علمی‌تر از این ویژگی‌ها شود. همان‌گونه که سالمی نیز بدان اشاره کرده است، این ویژگی‌های برگرفته از سه مجموعه مکمل از عوامل ایفا‌کننده نقش در دانشگاه‌های برتر جهان است که عبارت‌اند از:

- الف. تمرکز بسیار بر استعداد [شامل دانشجویان و استادان]؛
 - ب. بهره‌مندی از منابع فراوان [به منظور فراهم ساختن یک محیط مستعد یادگیری و انجام یافتن پژوهش‌های پیشرفته]؛
 - ج. مدیریت مطلوب [که موجبات تشویق در پیگیری چشم‌انداز راهبردی، خلاقیت و انعطاف‌پذیری را فراهم کند].
- سه عامل برشمرده مهم‌ترین زمینه‌ها و عوامل اصلی شکل‌دهنده دانشگاه‌های برتر جهان است که در ادامه در خصوص هر یک توضیحاتی ارائه شده است.

تمرکز بسیار بر استعداد [شامل دانشجویان و استادان]

در نظر برخی مهم‌ترین عامل در برتری و رجحان یک دانشگاه حضور تعداد زیادی از دانشجویان و اعضای هیأت علمی با استعداد و برجسته در آن دانشگاه است. دانشگاه‌های برتر جهان قادر به انتخاب بهترین دانشجویان و نیز ایجاد جاذبه برای مستعدترین استادان و محققان هستند. از این‌رو، برخی از این دانشگاه‌ها دستورالعمل‌هایی را با توجه به توانایی‌ها و امتیازاتشان دنبال

هیأت علمی ای که در رقابت برای جذب سرمایه‌گذاری‌های پژوهشی دولتی موفق هستند.

در کشور آمریکا حداقل دوسوم منابع سرمایه‌گذاری‌های دولتی در امر پژوهش توسط دانشگاه‌های برتر جهان جذب می‌شود. همچنین، در این کشور ۳/۲ درصد از GDP^{۱۵} مربوط به هزینه‌های آموزش عالی است. این در حالی است که در اروپا این میزان به ۱/۳ درصد از GDP بالغ می‌شود. هزینه سرانه دانشجویی در کشور آمریکا ۵۴ هزار دلار و در اروپا ۱۳/۵۰۰ دلار است. از سوی دیگر، تحقیقات نشان می‌دهد که سطح سرمایه‌گذاری‌ها در امور پژوهشی دانشگاهی همبستگی زیادی با رتبه‌های دانشگاهی در سطح بین‌الملل دارد. برای مثال، همچنان که سطح سرمایه‌گذاری‌ها در امور پژوهشی در دانشگاه‌های انگلستان و سوئیس افزایش یافته، رتبه‌های جهانی دانشگاه‌های آن‌ها نیز رشد داشته است. اما در کشورهای فرانسه و آلمان که این کار صورت نگرفته است، دانشگاه‌های آنان با رکود یا تنزل در رتبه‌های دانشگاهی مؤسسات رتبه‌بندی بین‌المللی مواجه شده‌اند [۱۳].

راه‌اندازی دانشگاه علم و فناوری ملک عبدالله با ۱۰ میلیارد دلار هزینه از این نوع سرمایه‌گذاری‌هاست [۷]. برای تجهیز آزمایشگاه‌های این دانشگاه ۱/۵ میلیارد دلار هزینه شده است. دانشگاه را یک هیأت امنای بین‌المللی و خارج از قوانین وزارت آموزش عالی این کشور اداره خواهد کرد. این دانشگاه در نظر دارد ۲۰۰۰ دانشجوی تحصیلات تکمیلی و ۱۰۰۰ پژوهشگر فوق دکتری را از میان نخبگان داخلی و به ویژه خارجی با پرداخت هزینه تحصیل آن‌ها جذب کند.

وفور منابع همچنین، سبب شده است که دانشگاه‌های برتر جهان قادر به جذب استادان و پژوهشگران برتر باشند. تحقیقات در خصوص حقوق اعطایی به پژوهشگران نشان می‌دهد که دانشگاه‌های خصوصی به‌طور متوسط ۳۰ درصد بیشتر از دانشگاه‌های دولتی به کادر علمی خود حقوق پرداخت می‌کنند. طی ۲۵ سال گذشته این فاصله دائماً در حال افزایش بوده است، به طوری که امروزه متوسط حقوق سالیانه ۱۰۶/۵۰۰ دلار در دانشگاه‌های دولتی آمریکا فقط ۷۸ درصد از حقوق یک استاد تمام در دانشگاه‌های خصوصی است. بنابراین، تعجب‌آور نیست که هیچ یک از دانشگاه‌های دولتی آمریکا جزو ۲۰ دانشگاه رتبه اول (دانشگاه‌های برتر جهان) قرار ندارند [۱۸].

مدیریت مطلوب [که موجب تشویق در پیگیری چشم‌انداز راهبردی، خلاقیت و انعطاف‌پذیری می‌شود]

سومین بعد از ویژگی‌های دانشگاه‌های برتر جهان ایجاد محیطی رقابتی، آزادی علمی و خودمختاری دانشگاهی است که این قبیل دانشگاه‌ها از آن به نحو مطلوب برخوردارند. مجله اکونومیست در سال ۲۰۰۵ نظام آموزش عالی کشور ایالات متحده آمریکا را به

عنوان بهترین نوع در جهان یاد می‌کند. این مجله دلیل این امر را نه به سبب ثروت این کشور، بلکه به واسطه استقلال نسبی نظام آموزش عالی آمریکا از دولت، روح رقابتی که حاکم بر همه ابعاد نظام آن است و توانایی این نظام در ایجاد کار و تولید علمی مرتبط و سودمند برای جامعه می‌داند. در این گزارش تأکید شده است که فضا و محیط در دانشگاه‌های این چنین نظامی، مساله رقابت را پرورش می‌دهد، تحقیقات علمی را آزاد می‌سازد و تفکر انتقادی، خلاقیت و نوآوری را تشویق می‌کند. به علاوه، مؤسسات آموزش عالی بهره‌مند از استقلال انعطاف‌پذیرترند، زیرا آن‌ها گرفتار مشکلات پیچیده دیوان‌سالاری و نیز استانداردهای تحمیلی خارج از دانشگاه نیستند و حتی سازوکارهای قانونی پاسخگویی آن‌ها را در قید و بند قرار نمی‌دهد. در نتیجه، این نوع دانشگاه‌ها قادر خواهند بود که منابع خود را با هوشمندی اداره کنند و با سرعت به ابعاد در حال تغییر بازار پاسخ دهند.

۸. نتیجه‌گیری

امروزه، نوعی تعامل دو طرفه بین مفهوم توسعه یافتگی و پیشرفت و برخورداری از دانشگاه‌های برتر جهان در کشورها وجود دارد. به تعبیر دیگر، هم توسعه‌یافتگی مشوق بهره‌مندی از این قبیل دانشگاه‌ها می‌شود و هم خدمات اجتماعی این نوع دانشگاه‌ها در امر پیشرفت کشورها مؤثر است. پدیده دانشگاه‌های برتر جهان امروزه برای بسیاری از کشورها به عنوان هدف و آرمان به شمار می‌آید. کشورهای جهان دریافته‌اند که اگر چه نیل به رتبه چنین دانشگاه‌هایی شاید در زمان کوتاه امکان‌پذیر نباشد، اما به دلیل فرصت‌هایی که از حرکت در جهت نیل به آن وجود دارد، برنامه‌ریزی‌هایی را برای این کار آغاز کرده‌اند. رتبه‌بندی‌های بین‌المللی دانشگاه‌ها پاسخگویی، رقابت و مدیریت داده‌ها و اطلاعات در باره دانشگاه‌ها را افزایش داده‌اند، اما ابهام در روش‌شناسی و ساده‌سازی فعالیت‌های جامع دانشگاهی از جمله نقاط ضعف آن‌هاست. همچنین، باید در نظر داشت که هدف اصلی رتبه‌بندی‌هایی که مجلات انجام می‌دهند، فروش بیشتر و سودآوری است. به هر حال، رتبه‌بندی مؤسسات و مجلات واقعی بین‌المللی است که بسیاری از تصمیمات جامعه دانشگاهی و ذینفعان را تحت تأثیر قرار داده است.

موضوع وضعیت دانشگاه‌های کشورهای اسلامی و ضرورت ارتقای جایگاه آن در رتبه‌بندی‌های جهانی برای نخستین بار در نشست سال ۲۰۰۵ کنفرانس اسلامی در مکه مطرح شد. در این نشست عنوان شد که در میان ۵۰۰ دانشگاه برتر دنیا تعداد بسیار اندکی از دانشگاه‌های جهان اسلامی قرار دارند که ایده رتبه‌بندی دانشگاه‌های عضو OIC از همان جا شکل گرفت. بدین‌سان که سازوکاری طراحی شد که توسط آن ۲۰ دانشگاه برتر جهان

15- Gross Domestic Product

6-Hazelkorn, Ellen, "Questions Abound as the College-Rankings Race Goes Global", Chronicle of Higher Education, March 13, 2011.

7-Lindsey, Ursula, "Saudi Arabia's \$10-Billion Experiment Is Ready for Results", Chronicle of Higher Education, June 26, 2011.

8-Liu, Nian CAI, "Building up World-Class Universities: A Comparison", Presentation at Research Institute for Higher Education, Hiroshima University, February 2009.

9- Liu, N. C. and Y. Cheng, "The Academic Ranking of World Universities: Methodologies and Problems, Higher Education in Europe 30, (2 July), PP.127-36, 2005.

10-Messer, Ron, "Building a World Class University", Business Intelligence Journal, January, PP.79-81.2009.

11-Richards, Alex and Coddington, Ron, "30 Ways to Rate a College", Chornicle of Higher Education, August 29, 2010.

12-Sadlak, Jan and CAI, Liu Nian, "World-Class: Aspirations and Reality Checks, Unesco European Center for Higher Education, 13-23, 2009.

13-Salmi, Jamil, The Challenge of Establishing World-Class Universities, World Bank Publication. Washington, P. 26, 2009.

14-Sternler, Steve, "An Overview of Content Analysis", Practical Assessment, Research and Evaluation, Internet: <http://pareonline.net/genpare.asp?Wh=0&abt=7>, 2006.

15-Times Higher Education, "The World University Tanking 2010", Available at: www.timeshighereducation.co.uk/world-university-rankings/2010-2011/top-200.html, July 27, 2011.

16-Times Higher Education. "Top 50 Engineering & Technology Universities", Available at: www.timeshighereducation.co.uk/world-university-rankings/2011-2012/engineering.html, January 3, 2012.

17-U.S. News and Report, "Best Colleges 2009", Available at: <http://colleges.usnews.rankingsandreviews.com/college/national-search>, December 20, 2010

18-U.S. News and Report, "America's Best Graduate Schools", Available at: <http://grad-schools.usnews.rankingsandreviews.com/grad>, June 10, 2008.

19-Weber, Robert P., Basic Content Analysis, 2nd ed., Newbury Park, PP. 44, 1990.

۲۰ - اوحدی، وحیدرضا، "معیارهای رتبه‌بندی دانشگاه‌ها"، مجله رهیافت، شماره ۴۱، پاییز و زمستان ۱۳۸۶.

۲۱ - بازرگان، عباس، «فرهنگ کیفیت و نقش آن در دستیابی به عملکرد مطلوب دانشگاه‌ها»، فصلنامه آموزش مهندسی ایران، سال سیزدهم، شماره ۵۰، تابستان ۱۳۹۰.

۲۲ - غفاری، محمد مهدی و مریم پاکپور، «نقش آینده نگر فناوری در نظام علم و آموزش مهندسی ژاپن»، فصلنامه آموزش مهندسی ایران، سال نهم، شماره ۳۶، زمستان ۱۳۸۶.

۲۳ - زارع بنادکوکلی، محمدرضا و محمد صالح اولیاء، "جایگاه علمی کشور ایران: مطالعه موردی پژوهش‌های مهندسی صنایع"، فصلنامه آموزش مهندسی ایران، سال سیزدهم، شماره ۵۰، تابستان ۱۳۹۰.

اسلام شناخته شوند و سپس، با حمایت‌هایی که از این دانشگاه‌ها می‌شود، بتوان آن‌ها را در فهرست ۵۰۰ دانشگاه برتر جهان قرار داد.

در ایران نیز رتبه‌بندی‌های بین‌المللی و ارتقای جایگاه دانشگاه‌های کشور مورد توجه مسئولان و رسانه‌ها قرار گرفته است. طی دو سال گذشته فقط دانشگاه تهران در بین دانشگاه‌های تراز اول جهان قرار گرفته است، اما برای اولین بار نام دانشگاه صنعتی شریف به عنوان تنها دانشگاه ایران در رتبه‌بندی سال ۲۰۱۱ مؤسسه تایمز در بین رتبه‌های ۳۰۰ تا ۳۵۰ و دانشگاه تهران در رده ۳۰۱ تا ۴۰۰ مؤسسه شانگ‌های در همین سال قرار گرفته است. نیل به این جایگاه به واسطه تلاش‌های صورت گرفته در جهت تولید علم در کشور طی سالیان اخیر بوده است. برای مثال، در بررسی وضعیت پژوهش علمی کشور ایران در مقایسه با سایر کشورهای جهان اسلام مشخص شد که ایران در سال ۱۳۸۶ از نظر میزان مقاله‌های علمی بین‌المللی در بین کشورهای اسلامی در ده سال جایگاه سوم و در سه سال منتهی به سال ۱۳۸۶ جایگاه دوم را دارا بوده است [۲۳].

به هر حال، شایسته است که مأموریت کلی دانشگاه‌های ایران در ارائه خدمات اجتماعی برای پیشرفت همه جانبه کشور خلاصه شود و دارای بیشترین اولویت باشد. با توجه به وفور استعداد‌های ممتاز در سطح دانشجویان و استادان، با افزایش بهره‌مندی دانشگاه‌ها از منابع حمایتی در کنار انجام یافتن اصلاحات اداری و مدیریتی به ویژه استقلال دانشگاه‌ها، می‌توان اقدامات مناسبی را برای خدمات اجتماعی با کیفیت آغاز کرد که این گونه اقدامات جای گرفتن دانشگاه‌های برتر کشور در رتبه‌بندی جهانی را به ارمغان خواهد آورد؛ اقداماتی که ضامن آینده‌ای روشن برای نظام آموزش عالی و تعالی آموزش، پژوهش و خدمات خواهد بود.

مراجع

- 1- ARWU, «Academic Ranking of World Universities - 2010», Available at: www.arwu.org/ARWU2010/jsp#, July.20, 201.
- 2-Altbach, Philip, "World Class Universities", Academe, PP 21-23, 2004.
- 3- Altbach, G. Philip, «The Costs and Benefits of World-Class Universities», An American's Perspective», Hong Kong America Center, Chinese University of Hong Kong, Hong Kong, China. 2003.
- 4- Deng, Qiaoqiao, Wang, Qi, and Liu, Nian Cai, National Initiatives for Building World Class Universities, Shanghai University, China, 2009.
- 5- Harvard University Financial Report (2010), Available at: http://cdn.wds.harvard.edu/fad/2010_full_fin_report.pdf, July 14, 2011.

مهندسان این کار را برای درگیر نشدن در بحث‌های مربوط به تقابل tab و space انجام دادند.

۱۹۵۷ جان بکوس و شرکت آی بی ام زبان فرتن را به وجود آوردند. چیز خنده‌داری در مورد آی بی ام و فرتن وجود ندارد. نوشتن برنامه فرتن بدون این که کراوات آبی (یونیفورم کارمندان آی بی ام) زده باشید باعث خطای نحوی^۵ می‌شود.

۱۹۵۸ جان مک کارتی و پل گراهام زبان لیسپ را اختراع کردند. به دلیل کمبود ذخایر پرانتز در دوران پس از جنگ، زبان لیسپ محبوبیت چندانی پیدا نکرد.

۱۹۵۹ گریس هاپر^۶ و چند سادیست (کسی که از زجر دیگران لذت می‌برد) دیگر زبان کوبول را اختراع کردند.

۱۹۶۴ جان کیمنی و توماس کورتز زبان بیسیک را به وجود آوردند، یک زبان برنامه‌نویسی غیرساخت یافته برای دانشمندان غیر کامپیوتر.

۱۹۷۰ گای استیل و جرالده سوسمان زبان تابعی^۷ اسکیم را به وجود آوردند. کار آن‌ها به یک سری مقاله درباره حساب لاند انجامید. این مقالات محبوبیتی برای لاند‌ها به همراه نیاورد تا آن که زبان جلاوا با عدم استفاده از آن‌ها باعث محبوبیتشان شد.

۱۹۷۰ نیکلاس ویرث زبان پاسکال را به وجود آورد. پاسکال یک زبان رویه‌ای^۸ بود. بلافاصله انتقاداتی علیه پاسکال مطرح شد درباره این که چرا به جای این که مثل زبان C بنویسید $x=x+y$ در این زبان باید

یک تقسیم بر صفر

تاریخچه خلاصه، ناقص و نادرست زبان‌های برنامه‌نویسی

۱۸۰۱ ژوزف ماری ژاکارد از کارت‌های منگنه برای هدایت یک ماشین پارچه‌بافی استفاده کرد تا بر روی یک پارچه، عبارت «hello, world» را ببافد. کاربران آن زمان به دلیل عدم استفاده سیستم از عملیات بازگشتی^۱، پردازش موازی و انتزاع^۲، استقبال چندانی از آن نکردند.

۱۸۴۲ ایدا لاولیس^۳ نخستین برنامه کامپیوتری را نوشت. او در تلاش‌هایش با مانع کوچکی روبرو شد و آن این بود که هنوز هیچ کامپیوتری برای اجرای برنامه‌اش وجود نداشت. معماران سازمانی بعداً از تکنیک‌های او به منظور برنامه‌نویسی در UML استفاده کردند.

۱۹۳۶ آلن تورینگ تمام زبان‌های برنامه‌نویسی که تا ابد به وجود خواهند آمد را اختراع کرد اما پیش از آن که بتواند آن‌ها را به ثبت برساند از دنیا رفت.

۱۹۳۶ آلونزو چرچ نیز تمام زبان‌های برنامه‌نویسی که تا ابد به وجود خواهند آمد را اختراع کرد اما کارش بهتر از تورینگ بود. حساب لاندی^۴ او به دلیل آن که به اندازه کافی شبیه زبان C نبود نادیده گرفته شد. این بی‌اعتنایی و نادیده انگاری علیرغم این واقعیت صورت گرفت که زبان C هنوز اختراع نشده بود.

دهه ۱۹۴۰ کامپیوترهای متعددی مستقیماً از طریق سیم‌پیچی و استفاده از سوئیچ‌ها «برنامه‌نویسی» شدند.

5- syntax error
6- Grace Hopper
7-functional
8- procedural

1- recursion
2- abstraction
3- Ada Lovelace
4- lambda calculus

نوع اختلال یادگیری که مانع از درک شخص به هنگام خواندن می‌شود) بوده‌اند.

۱۹۸۷ لری وال خوابش برد و پیشنهادش روی صفحه کلید افتاد. وقتی بیدار شد به رشته حروفی که بر روی نمایشگرش بود نگاه کرد و به او الهام شد که این رشته، تصادفی نیست بلکه مثالی از برنامه‌ای به یک زبان برنامه‌نویسی است که خدا می‌خواهد او طراحی کند. بدین ترتیب، زبان پرل متولد شد.

۱۹۹۰ کمیته‌ای متشکل از سایمون پیتون جونز، پل هوداک، فیلیپ وادلر، اشتون کوچر، و افرادی از سازمان حمایت از حیوانات، زبان برنامه‌نویسی تابعی هسکل را به وجود آوردند.

۱۹۹۵ راسموس لردورف در یک رستوران ایتالیایی به این نتیجه رسید که بشقاب اسپاگتی‌اش مدل خوبی برای درک وب است و بنابراین کاربردهای تحت وب باید شبیه رسانه خود باشند. او سپس در پشت دستمال سفره‌اش زبان PHP را طراحی کرد. PHP مخفف Programmable Hyperlinked Pasta است. مستندات PHP تا امروز پشت همان دستمال سفره باقی مانده است.

۱۹۹۵ برندن ایک^{۱۳} تمام اشتباهاتی که در طراحی زبان‌های برنامه نویسی شده بود را جمع آوری کرد، چند اشتباه جدید نیز ابداع کرد و زبان لایواسکریپت^{۱۴} را به وجود آورد. بعداً با سوء استفاده از محبوبیت جاوا، نامش به جاوا اسکریپت تغییر داده شد. بعد از آن هم با سوء استفاده از محبوبیت بیماری‌های پوستی نام این زبان به ECMAScript تغییر یافت.

۱۹۹۶ جیمز گازلینگ زبان جاوا را اختراع کرد. جاوا یک زبان نسبتاً طول و دراز، بر پایه رده‌ها^{۱۵}، به صورت ایستا نوع بندی شده^{۱۶} و شیء گراست. شرکت سان مایکروسستمز با افتخار، ظهور جاوا را بشارت داد.

۱۹۹۷ آندرس هایلبرگ زبان C# را اختراع کرد. C# یک زبان نسبتاً طول و دراز، بر پایه رده‌ها، به صورت ایستا نوع بندی شده و شیء گراست. مایکروسافت با افتخار، ظهور C# را بشارت داد.

۲۰۰۳ مارتین اودرسکی با الهام از تبلیغ تلویزیونی بستنی شکلاتی، زبان اسکالا را به وجود آورد. زبانی که ساخت‌های زبان‌های شیء گرا و تابعی را یکپارچه کرده است. طرفداران زبان‌های شیء گرا و طرفداران زبان‌های تابعی بلافاصله اعلام جنگ کردند.

نوشت $x:=x+y$. این انتقاد علیرغم این که هنوز زبان C به وجود نیامده بود، مطرح گشت.

۱۹۷۲ دنیس ریچی یک اسلحه قوی و قدرتمند اختراع کرد که همزمان از جلو و عقب شلیک می‌کرد. او که از تعداد کشته‌ها و مجروح شدگان اسلحه‌اش راضی نشده بود C و یونیکس را اختراع کرد.

۱۹۷۲ آلن کولمرو زبان برنامه‌نویسی منطقی پرولوگ را طراحی کرد. هدف او ایجاد زبانی بود با هوشمندی یک بچه دوساله. او با نشان دادن اجرای یک برنامه پرولوگ که به هر پرس‌وجویی^۹ پاسخ «no» می‌داد ثابت کرد که به هدفش دست یافته است.

۱۹۷۳ رابین میلنر زبان ML را به وجود آورد، زبانی بر پایه نظریه M&M. بعداً زبان ML باعث به وجود آمدن زبان SML شد که دارای قواعد معنایی^{۱۰} تعریف شده صوری بود. از دیگر زبان‌های مشهور در خانواده ML می‌توان از OCaml، F# و ویژوال بیسیک نام برد.

۱۹۸۰ آلن کی زبان اسمال تاک و عبارت «شیء گرایی» را ابداع کرد. وقتی از او پرسیدند این چه معنایی می‌دهد پاسخ داد: «برنامه‌های اسمال تاک فقط اشیاء هستند.» وقتی از او پرسیدند اشیاء از چه چیزی به وجود می‌آیند؟ پاسخ داد: «اشیاء.» وقتی دوباره از او سؤال کردند گفت: «بیخود وقتتان را تلف نکنید. همه چیز در اینجا شیء است.»

۱۹۸۳ به افتخار ایدالولیس که برنامه‌هایی نوشته بود که هیچوقت اجرا نشده بودند، ژان ایشبیا^{۱۱} و وزارت دفاع آمریکا زبان برنامه‌نویسی ایدا را اختراع کردند. علیرغم کمبود شواهدی مبنی بر این که هیچ برنامه مهمی به زبان ایدا نوشته شده باشد اما این یک پروژه بسیار موفق بود زیرا چندین هزار پیمانکار بیکار و سرگردان را سرکار گذاشت.

۱۹۸۳ بیارنه استراستروپ هر چیزی که درباره C از هر کجا شنیده بود را روی هم ریخت و C++ را به وجود آورد. زبان حاصله آنقدر پیچیده بود که برنامه‌هایش را برای ترجمه باید به آینده حواله می‌کردند.

۱۹۸۶ بردکاکس و تام لاو زبان Objective-C را به وجود آوردند و اعلام کردند که این زبان امنیت حافظه C (!) و سرعت درخشان اسمال تاک (!!) را یکجا گرد آورده است. تاریخ دانان معاصر حدس می‌زنند که آن دو دچار خوانش پریشی^{۱۲} (یک

13- Brenden Eich

14- Live Script

14-class based

15-statically typed

16- statically typed

9- query

10- semantics

11-Jean Ichbiah

12-dyslexia



Gozareh - e Computer

Vol. 34, No. 203

Contents of Persian Section

ARTICLES

A Supervised Evolutionary Computer Aided Management Accounting System	8
Quantum Computing	14
Implementing e-Cities	24
A Survey on Querying Encrypted Data	53
A Secure Routing Protocol for Ad Hoc Networks	68
Data Warehouse and On Line Analytical Process	76

REPORTS

Open Educational Resources Declaration	12
16 th International AI and Signal Processing Conference	23
A Survey on Activities of Iranian Scientific Societies	35
C++0x Seminar	74
A Seminar on Computer Game Industry	75

DEPARTMENTS

News	2
Coders At Work (Part 13)	38
Viewpoint: Commemoration of Professor Parviz Shahriari	63
Book Review: Future Digital World	86
Glossary	94
Miscellaneous: Academic Ranking of World Universities	95
Entertainment: One Divide Zero	102

Board of Directors

- Ali Farrokhi
- Ali Reza Bagheri
- Abbas Nowzari Dalini
- Mohsen Sadighi Moshkenani
- Ali Reza Mahyar
- Ebrahim N. Mashayekh (President)
- Ebrahim Abtahi (vice- President)
- Mohammad Sanati (Secretary)
- Mohammad Hassan Mehvari (Treasurer)

Committees

- Public Relation: Mohammad Hassan Mehvari
- Membership: Mohammad Hassan Mehvari
- Publishing: Ebrahim N. Mashayekh
- Science and Technology: Ebrahim Abtahi
- Education: Ali Farrokhi



Vol. 34, No. 203

July, 2012

Chief Editor

Ebrahim N. Mashayekh

Circulation: CR is published bimonthly by ISI. Please address your subscription requests to: Anoosh Hosseini, P.O.Box 61622, Sunnyvale, CA 94088 USA. annoosh @ gpg.com. Annual subscription is include in membership fee. Non-member price: US\$50 per year (6 issues). CR features original and translated articles, news and reviews on all aspects of computing in Iran and abroad.

Submissions: Submit your articles to: The Editor, *Computer Report*, P.O.Box 1196, Tehran 14155, IRAN, mashayekh@isi.org.ir. All submissions are subject to editing for style, clarity and space consideration.

Editorial: Unless otherwise stated, articles and reports reflect the author's opinion. Inclusion does not imply endorsement by ISI.

Mailing List Rental: ISI lists are available for computer-related products and services.

Copyright © 2011 by Informatics Society of Iran. Copying without fee is permitted with credit to the source.