Chapter 14
Current trends in system development

Asst.Prof.Dr. Supakit Nootyaskool

Faculty of Information Technology

King Mongkut’s Institute of Technology Ladkrabang

22 November 2019

Outline

- Trends in System Development Methodologies
- Trends in Technology Infrastructure
- Trends in Application Software Availability
- The Web as an Application Platform

Objective

- Describe the elements of the Unified Process (UP)
- Compare and contrast the features of Extreme Programming and Scrum development
- Describe the major trends in devices, connectivity, Internet, and telephone technologies
- List and describe the various methods of deploying application software
- List and describe the various elements that enhance Web applications
- Describe the various approaches to developing Rich Internet Applications (RIAs)

Overview

- This book has focused on teaching you the processes and skills associated with a system development project emphasizing agile, iterative development using UML
- Three very specific system development methodologies are described in this chapter: Unified Process (UP), Scrum, and Extreme Programming (XP)
- Additionally, three important trends affecting system development are discussed: consumer devices, the distribution of application software, and the continuing movement toward Web-based applications

Trends in system development methodologies

- Throughout this text we have emphasized Agile, iterative development
- We have also emphasized traditional project management techniques (updated for iterative development) and other widely used and up to date planning, analysis, design, implementation, and deployment concepts and techniques
- UML models and modeling techniques have been used throughout
- The overall approach used (the book's methodology) is very current in this regard, but it is generic

Three current methodologies

- There are many specific system development methodologies used in practice, and this chapter covers three influential ones
  - The Unified Process (UP)
  - Extreme Programming (XP)
  - Scrum
- These methodologies all use an iterative SDLC
- The Unified Process uses UML
- Extreme Programming and Scrum are based on Agile principles, but the Unified Process can also be used in an agile fashion
- Many organizations mix and match features of each of these when creating their own methodology

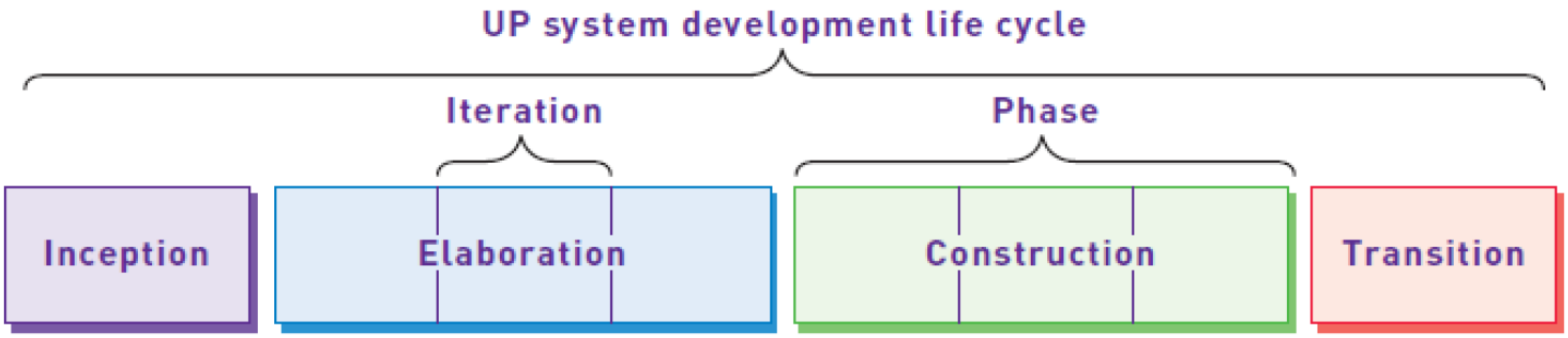
Methodology: *a system of ways of doing, teaching or studying*
Principle: *a basic idea or rule that explain or controls how something happened or works*

The unified process (UP)

- Originally developed by Booch, Rumbaugh, and Jacobson, who previously developed UML at Rational Software (now part of IBM)
- The UP is now widely recognized as a highly influential innovation in software development methodologies for object-oriented development using an adaptive approach
- The original version of UP defined an elaborate set of activities and deliverables for every step of the development process
- More recent versions are streamlined, with fewer activities and deliverables, simplifying the methodology
- Much of the book’s methodology is loosely based on the Unified Process (in an agile form)

The unified process: life cycle

- The Unified Process Life Cycle model includes iterations and phases (the SDLC in this text is very similar, but left out the UP phases for simplicity)
- Each UP phase is made up of iterations. The phases are Inception, Elaboration, Construction, and Transition

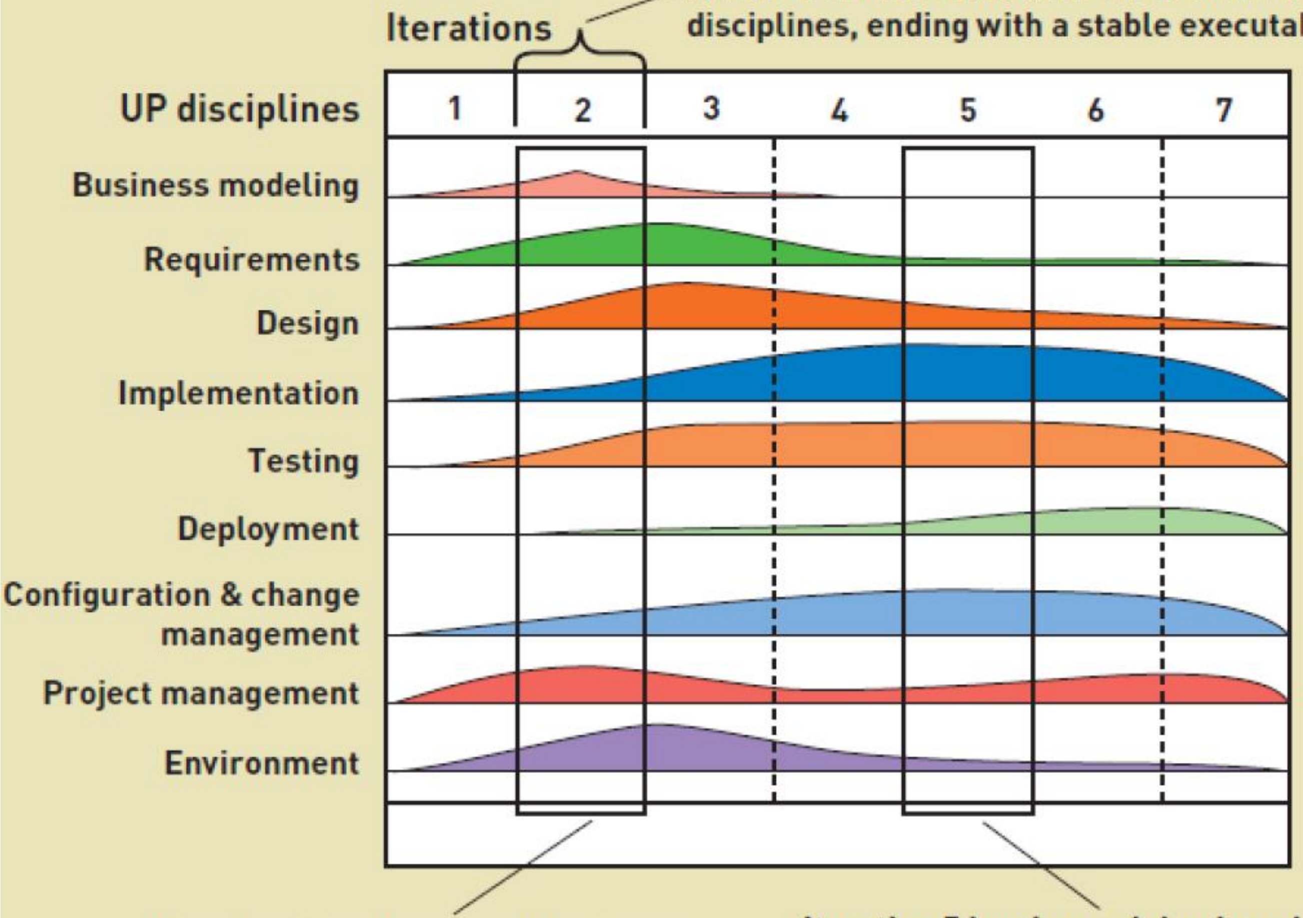


The unified process: disciplines

- UP Disciplines – a set of functionally related activities that combine to enable the development process in a UP project (each like a core development process):
  - Business modeling
  - Requirements
  - Design
  - Implementation
  - Testing
  - Deployment
  - Configuration and change management
  - Project management
  - Environment

Discipline (training): Training which procedures or self-control, often in the form of rules

The unified process: disciplines in each iteration

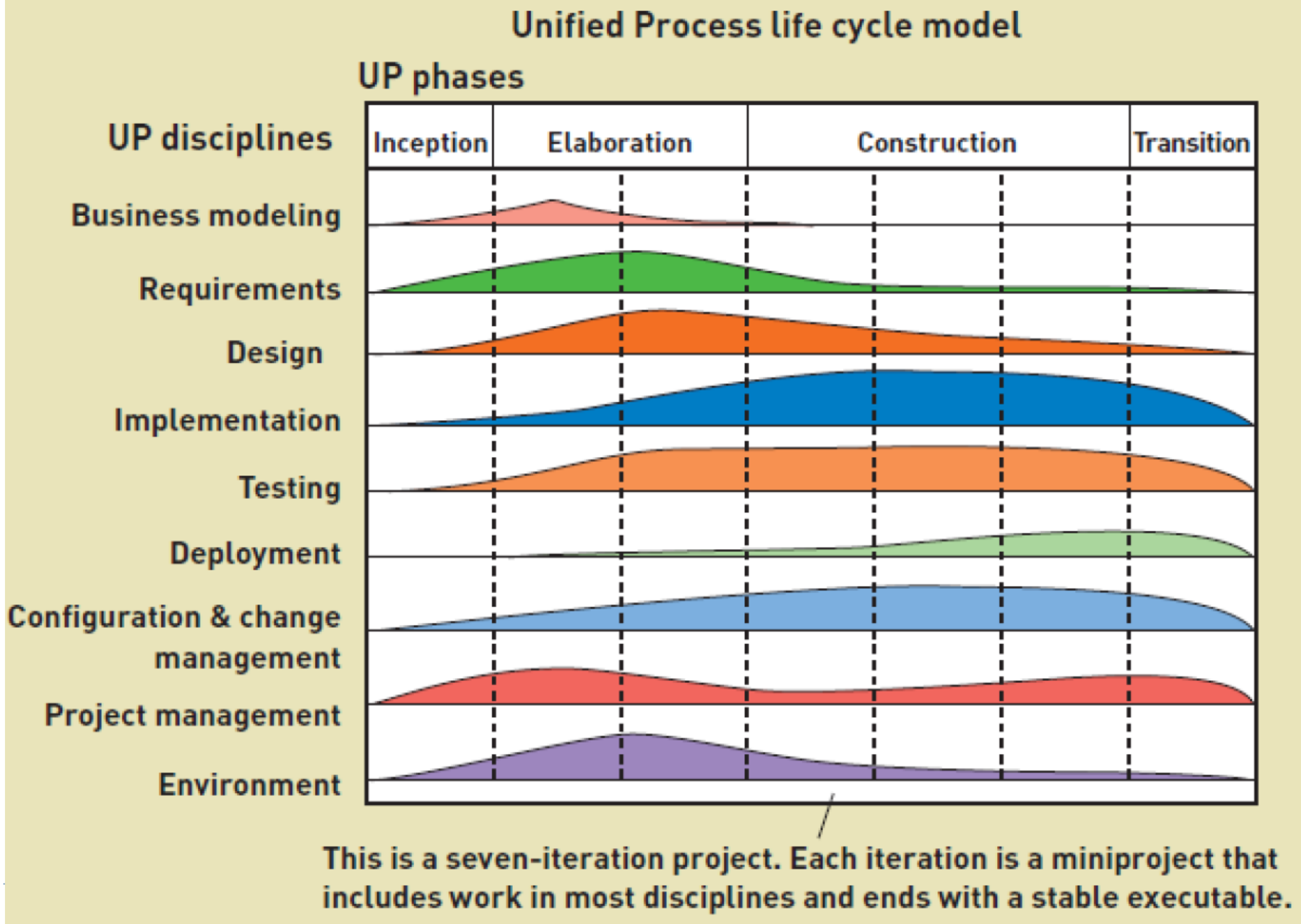


A four-week iteration includes work in most disciplines, ending with a stable executable.

Iteration 2 involves much business modeling and requirements but still includes some design, implementation, and testing activities.

Iteration 5 involves minimal modeling and requirements, some design, but much more implementation and many more testing activities.

The unified process: complete UP SDLC model



This is a seven-iteration project. Each iteration is a miniproject that includes work in most disciplines and ends with a stable executable.

The unified process: Philosophies/Practices

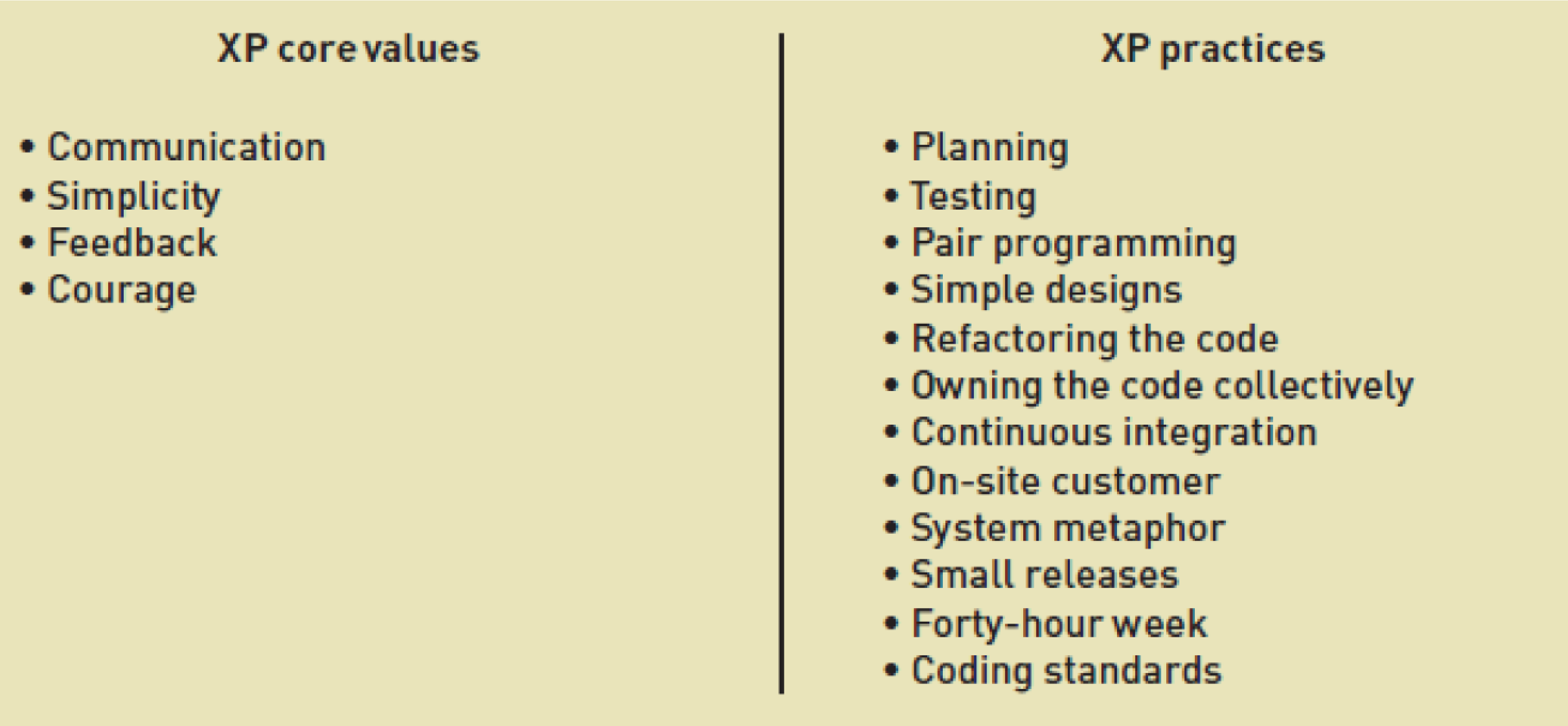
- Focuses early and often on users
- Use case driven
- Model driven – uses UML exclusively
- Iterative, but provides management structure by defining phases
- Focuses on defining an architecture
- Adaptable to needs of a specific project
- Can be made highly agile, but originally had heavy ceremony

Extreme programming (XP)

- One of the original agile development methodologies (perhaps a reaction to the original UP) from Kent Beck
- "Extreme" often thought to be radical, but really just focuses intently on industry best practices and combines best practices in new ways
- XP is based on core values
  - Communication
  - Simplicity
  - Feedback
  - Courage
- XP also defines a set of XP practices

Radical (supporting change): a person who supports great social and political change.
Courage: the ability to control your fear in dangerous or difficult situation.

Extreme programming (XP) core value and practices



XP core values

- Communication
- Simplicity
- Feedback
- Courage

XP practices

- Planning
- Testing
- Pair programming
- Simple designs
- Refactoring the code
- Owning the code collectively
- Continuous integration
- On-site customer
- System metaphor
- Small releases
- Forty-hour week
- Coding standards

Extreme programming (XP):core value

- Communication—one of the major causes of project failure is a lack of open communication among the right players at the right time and at the right level
- Simplicity—XP includes techniques to reinforce keeping things simple to make it a standard way of developing systems
- Feedback—as with simplicity, getting frequent, meaningful feedback is recognized as a best practice of software development
- Courage—developers always need courage to face the harsh choice of doing things right or throwing away bad code and starting over

Extreme programming (XP):XP practices

- Planning—XP planning focuses on making a rough plan quickly and then refining it as things become clearer. This reflects the Agile development philosophical dictum that change is more important than detailed plans
- Testing—XP intensifies testing by requiring that the tests for each use case (story) be written first—before the solution is programmed
- Pair Programming—XP practice in which two programmers work together on designing, coding, and testing software
- Simple Designs—XP conforms to the principles of Agile Modeling. It accomplishes the desired result with as few classes and methods as possible and that doesn't duplicate code

Extreme programming (XP):XP practices (2)

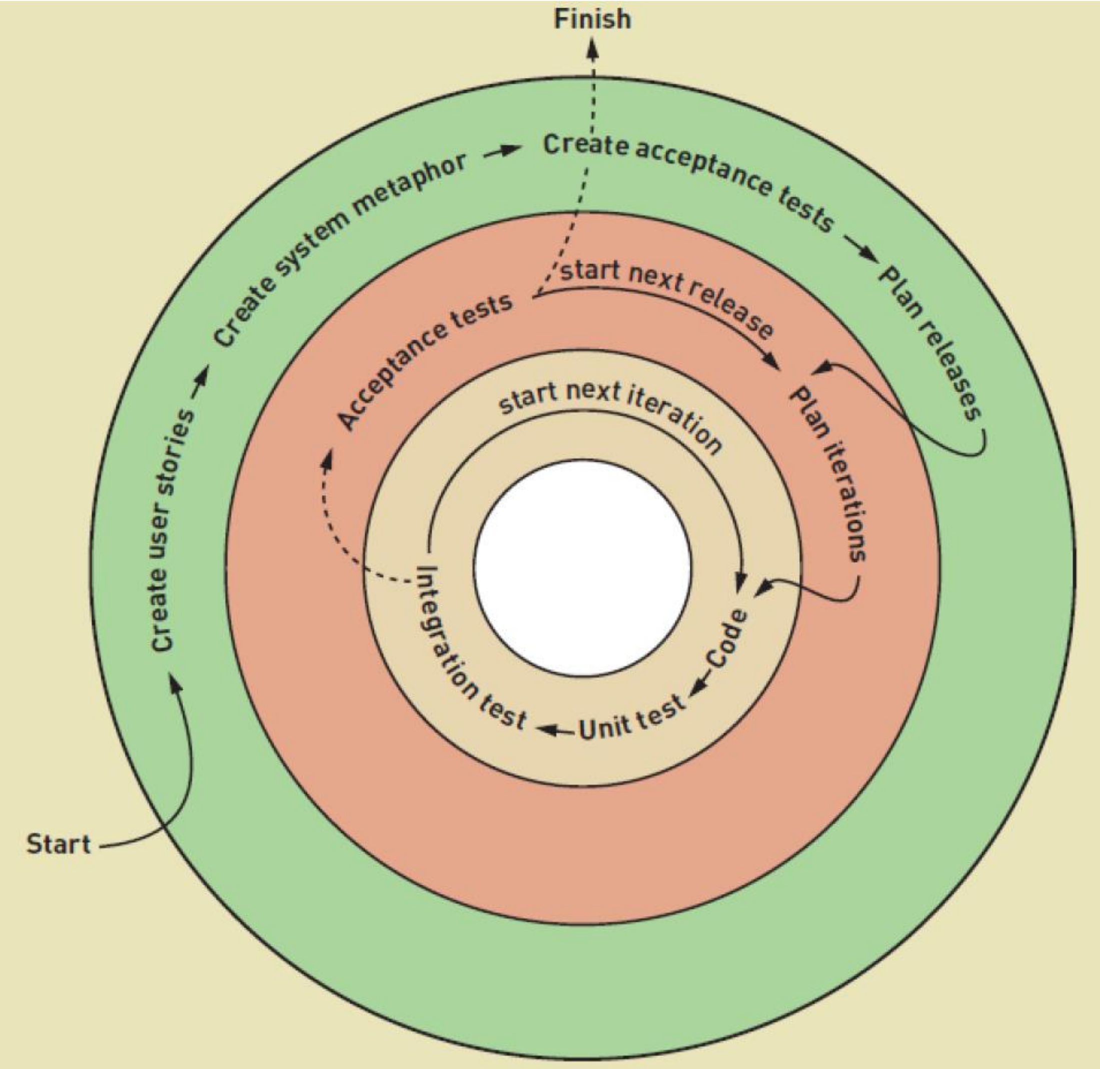
- Refactoring the Code— refactoring is the technique of improving the code without changing what it does. XP programmers continually refectory their code to achieve a simpler design
- Owning the Code Collectively —in XP, everyone is responsible for the code. Collective ownership allows anyone to modify any piece of code.
- Continuous Integration —this practice embodies XP's idea of "growing" the software. Small pieces of code—which have passed the unit tests—are integrated into the system daily or even more often
- On-Site Customer —as with all adaptive approaches, XP projects require continual involvement of users who can make business decisions about functionality and scope

Extreme programming (XP):XP practices (3)

- System Metaphor —a system metaphor should be easily understood and well known to the members of the development team. It can guide members toward a vision and help them understand the system
- Small Releases —consistent with the entire philosophy of growing the software, small and frequent releases provide upgraded solutions to the users and keep them involved in the project
- Forty-Hour Week — the exact number of hours a developer works isn't the issue. The issue is that the project shouldn't be a death march that burns out every member of the team
- Coding Standards —developers should follow standards for coding and documentation

XP activates

- Project activates
- Release activates
- Iteration activates



Scrum

- Another influential agile, iterative development methodology based on ideas from Rugby

- A Scrum is used to get a ball back into play after a penalty--it begins quickly, is a very intense effort, involves the entire team, and usually only lasts for a short duration

- Scrum philosophy is the complete control a team experts over its own organization and its work processes. Software is developed incrementally, and controls are imposed empirically—by focusing on things that can be accomplished.

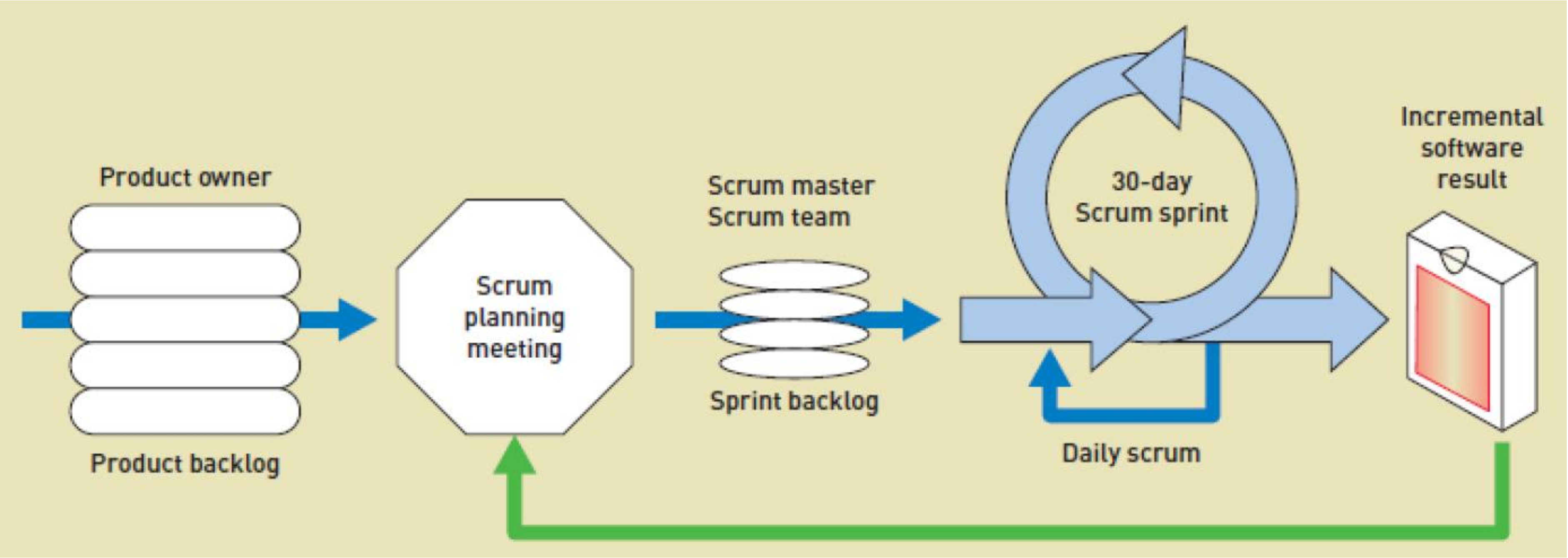
Scrum organization

- Product backlog – a prioritized list of user requirements used to choose work to be done in a Scrum project
  - Only a few of the high-priority items are worked on at a time
- Product owner – the client stakeholder for whom the system is being built
  - Responsible for project backlog and priorities
- Scrum master – the person in charge of a Scrum project—similar to a project manager
- Scrum team is usually 5 to 9 people
  - Scrum team sets own goals, organizes self, makes decisions

Scrum practices

- Sprint – a time-controlled mini-project that implements a specific portion of a system
  - Firm 30 day time box with specific goal or deliverable
  - The scope of that sprint is then frozen, and no one can change it—neither the product owner nor any other users
    - Sprint backlog defines the scope
- Daily Scrum – a daily meeting of all members of the team to report progress (15 minutes max)
- Sprint final half-day review meeting – scheduled to review and identify changes needed for the following sprints

Scrum development process



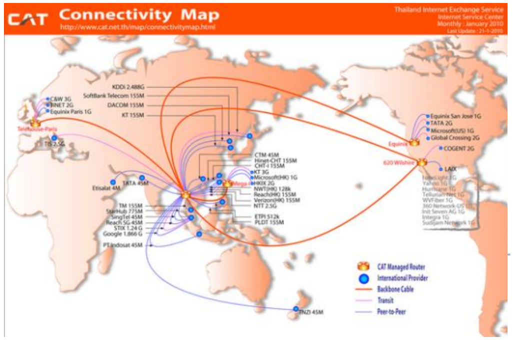
Trends in technology infrastructure

- New Client Computing Devices and Applications –Small, mobile, connected
  - Device-top application (Standalone application) – a computer application that is built to execute on a local device without requiring a client/server connection
  - Free-standing Internet application (Offline and connect internet sometime) –a client/server application that is self-contained and doesn't require the use of a Web browser
  - Browser-based application –a client/server application that executes locally within the control of a Web browser

Google Docs

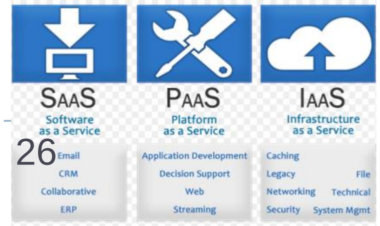
Trends in technology infrastructure

- Internet and Telephone Communications
  - Internet, the telephone, and television are merging
  - Long-distance telephone communication and long-distance Internet communication have very similar requirements and capability
  - Internet backbone – the primary data routes between large, strategically interconnected networks and routers on the Internet
  - Last mile – the final leg that delivers connectivity from the Internet network to the customer



Trends in technology infrastructure

- Back-End Computing –the server computers that provide the content—dynamic and static—for all applications that access servers through the Internet
  - As more and more client computers try to access a particular application, the workloads on the servers becomes incredibly heavy
  - Several factors are driving the need for large back-end computing services
    - Trend toward continuous connectivity to the Internet
    - Massive amount of data stored—many applications in the world of commerce keep a history of all the transactions that occur
    - The consolidation of processing in central locations



Data and programs are off the PC and back in massive data centers

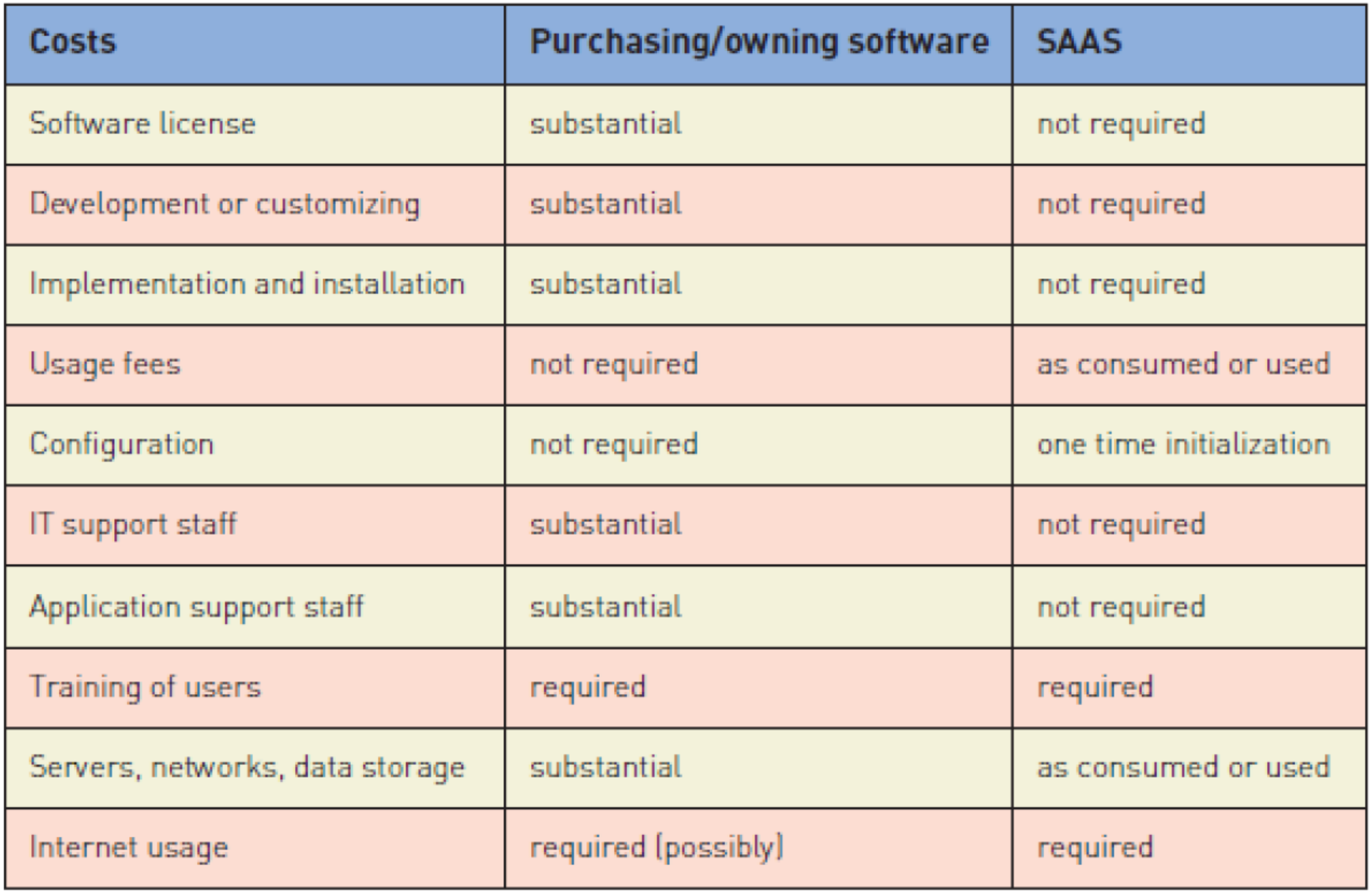
Trends in application software availability

- Software as a service (SAAS) – a software delivery model similar to a utility, in which the application and its associated data are accessed via the Internet without locally installed programs
- Open-source software – a method of developing, delivering, and licensing software that makes the application source code freely available to any interested developer or client

EDIT ONLINE

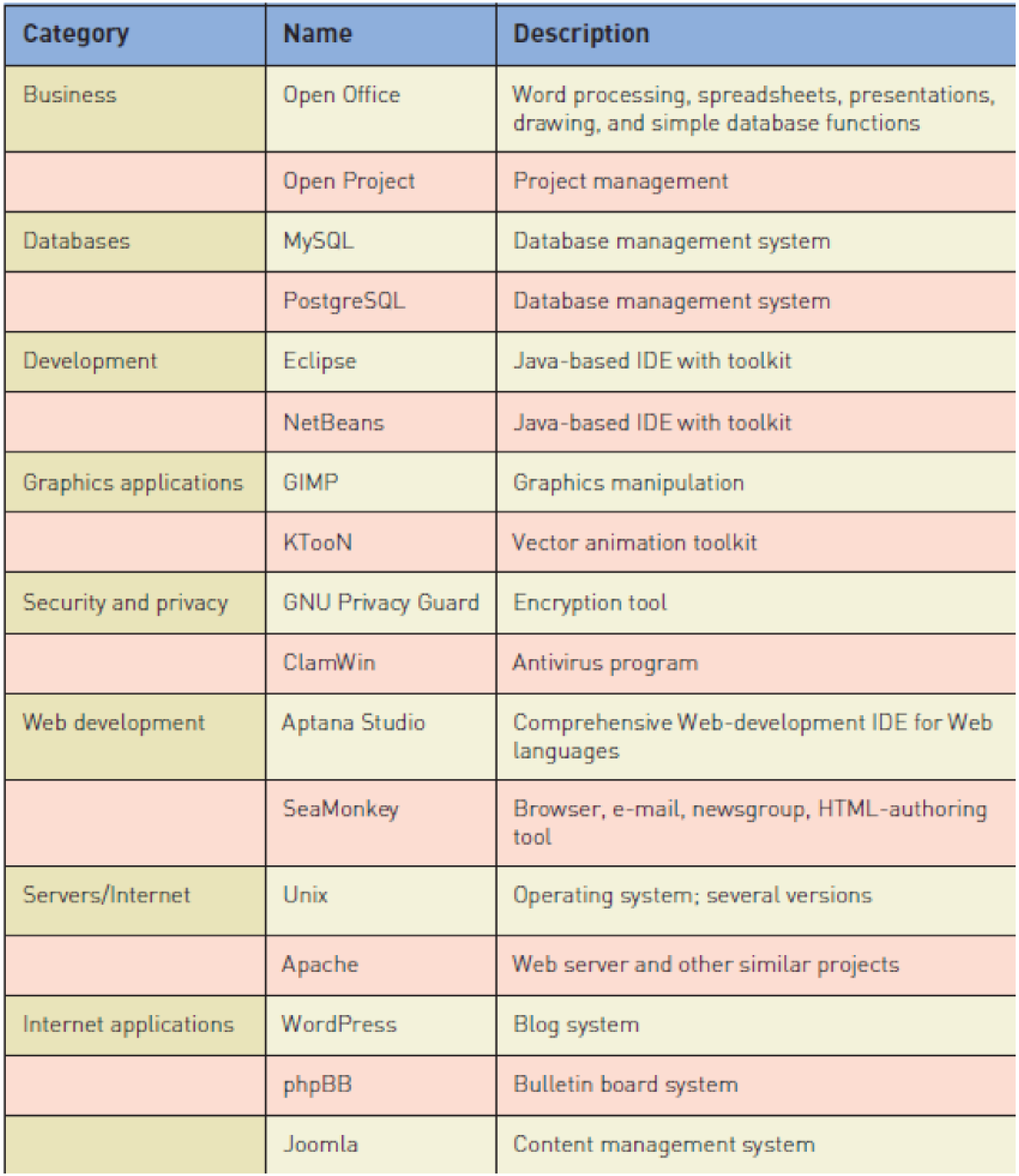
Use easy web tools to add quick tweaks and fun effects.

Owning software VS. SAAS



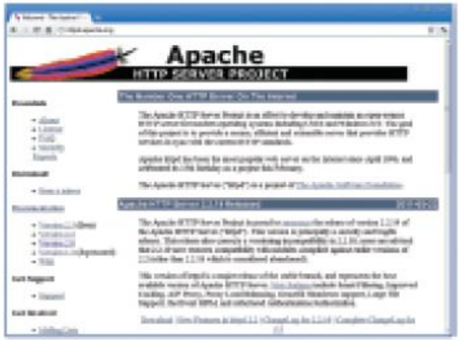
Costs	Purchasing/owning software	SAAS
Software license	substantial	not required
Development or customizing	substantial	not required
Implementation and installation	substantial	not required
Usage fees	not required	as consumed or used
Configuration	not required	one time initialization
IT support staff	substantial	not required
Application support staff	substantial	not required
Training of users	required	required
Servers, networks, data storage	substantial	as consumed or used
Internet usage	required (possibly)	required

Open source software applications

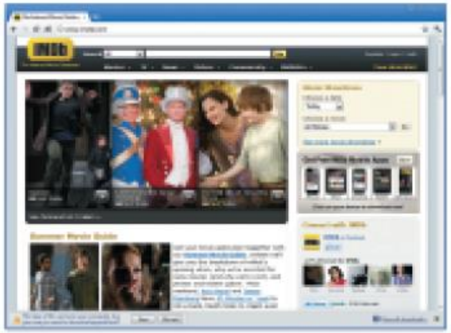


Category	Name	Description
Business	Open Office	Word processing, spreadsheets, presentations, drawing, and simple database functions
	Open Project	Project management
Databases	MySQL	Database management system
	PostgreSQL	Database management system
Development	Eclipse	Java-based IDE with toolkit
	NetBeans	Java-based IDE with toolkit
Graphics applications	GIMP	Graphics manipulation
	KTooN	Vector animation toolkit
Security and privacy	GNU Privacy Guard	Encryption tool
	ClamWin	Antivirus program
Web development	Aptana Studio	Comprehensive Web-development IDE for Web languages
	SeaMonkey	Browser, e-mail, newsgroup, HTML-authoring tool
Servers/Internet	Unix	Operating system; several versions
	Apache	Web server and other similar projects
Internet applications	WordPress	Blog system
	phpBB	Bulletin board system
	Joomla	Content management system

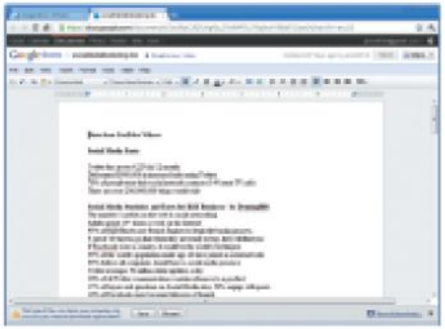
The web as an application platform—the evolution



1. Simple pages with text, static images, and links
   (e.g., httpd.apache.org)



2. Animated pages with media and plugins
   (e.g., www.imdb.com)



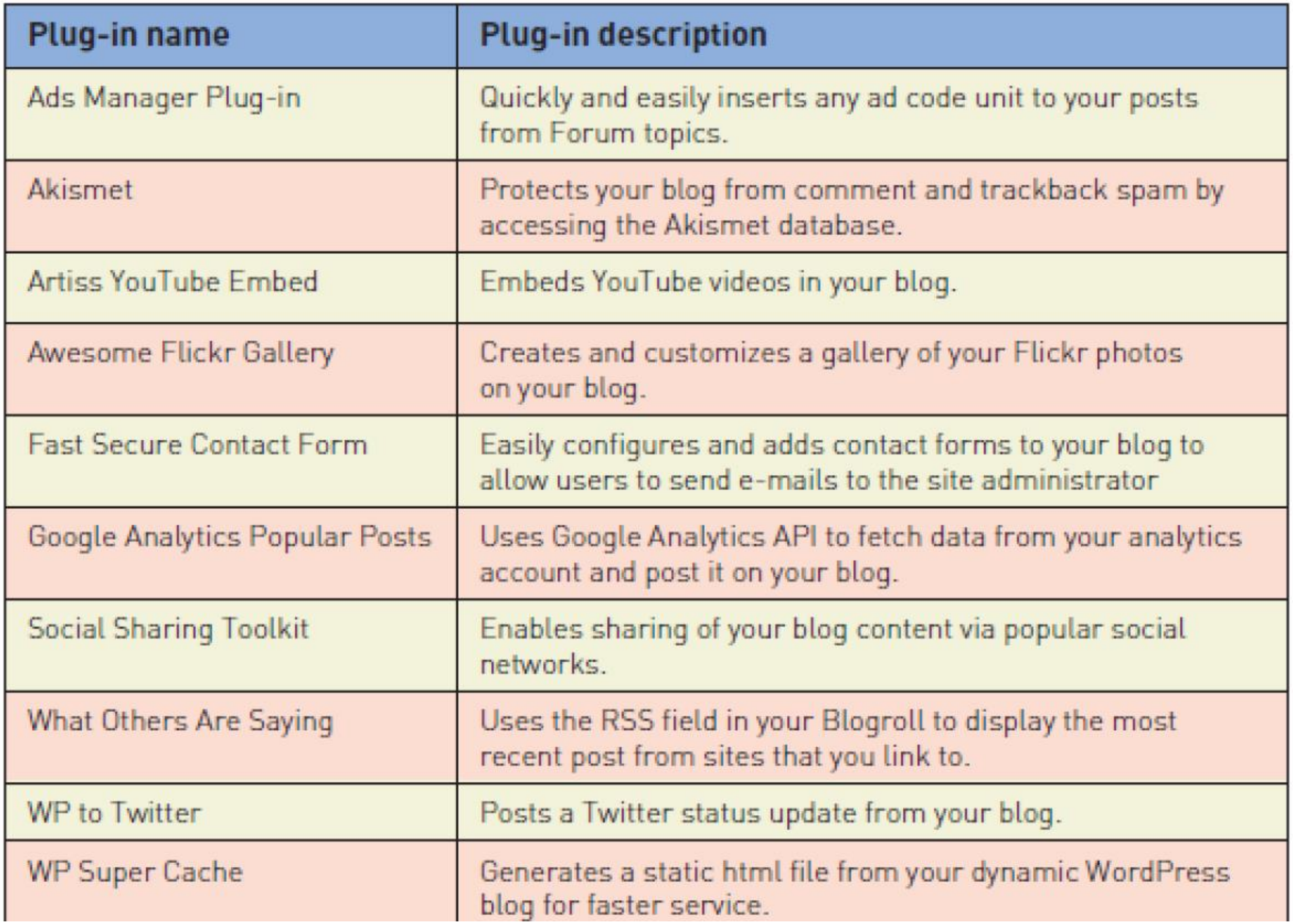
3. Rich Internet Applications with active user involvement
   (e.g., docs.google.com)

Courtesy of Apache, IMDB, and Google

The web as an application platform

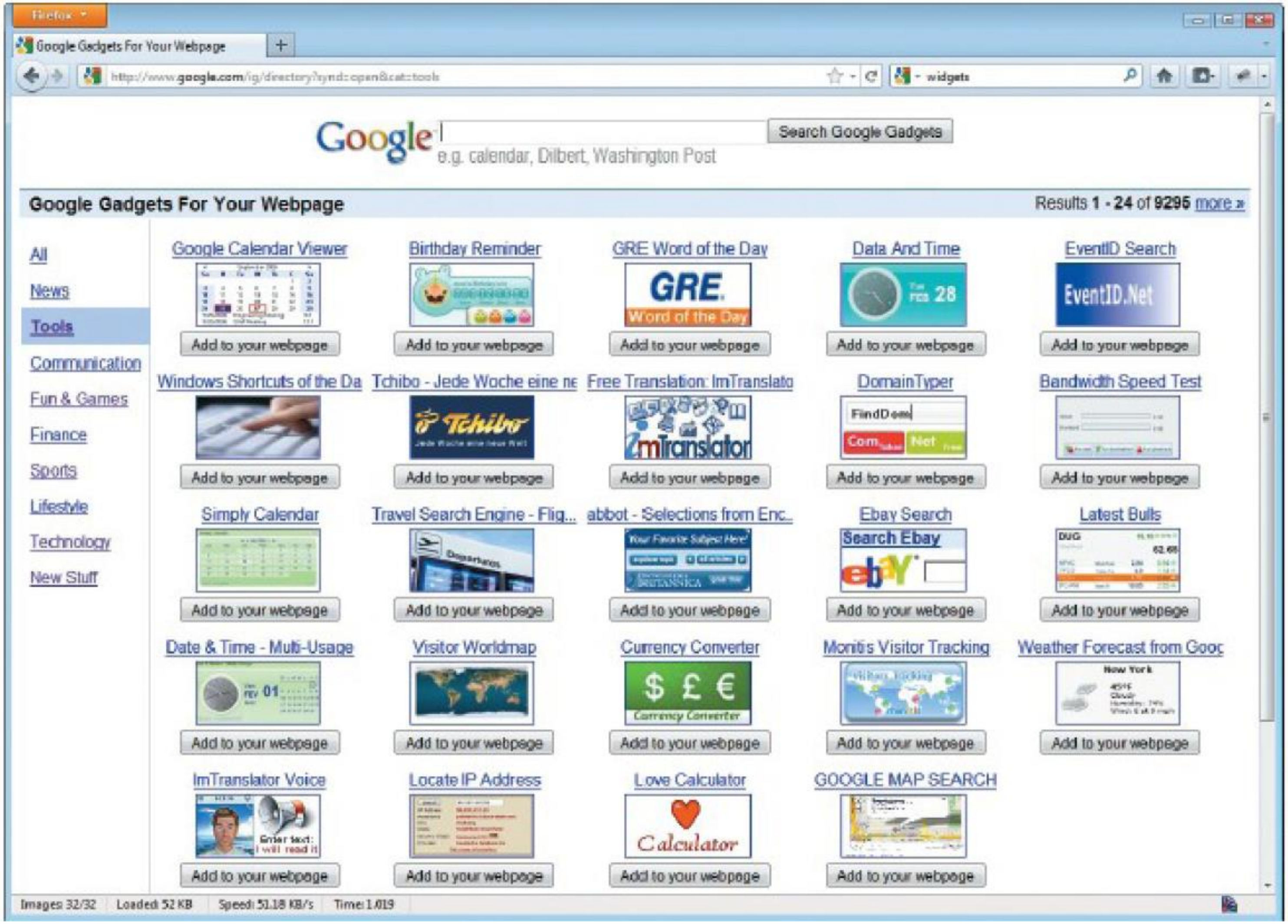
- Web 2.0 – a loosely defined, nonstandard term used to refer to Web sites that permit user-generated content and user interaction, such as social networking sites
- Plug-ins – a software component that adds specific capabilities to a larger software application
- Widget – a type of plug-in that focuses on enhancing the user interface with additional capability (also called gadget)
- Theme – a type of add-on to an application that allows the look and feel, such as colors and layout, to be changed

Sample plug-ins for WordPress blog



Plug-in name	Plug-in description
Ads Manager Plug-in	Quickly and easily inserts any ad code unit to your posts from Forum topics.
Akismet	Protects your blog from comment and trackback spam by accessing the Akismet database.
Artiss YouTube Embed	Embeds YouTube videos in your blog.
Awesome Flickr Gallery	Creates and customizes a gallery of your Flickr photos on your blog.
Fast Secure Contact Form	Easily configures and adds contact forms to your blog to allow users to send e-mails to the site administrator
Google Analytics Popular Posts	Uses Google Analytics API to fetch data from your analytics account and post it on your blog.
Social Sharing Toolkit	Enables sharing of your blog content via popular social networks.
What Others Are Saying	Uses the RSS field in your Blogroll to display the most recent post from sites that you link to.
WP to Twitter	Posts a Twitter status update from your blog.
WP Super Cache	Generates a static html file from your dynamic WordPress blog for faster service.

Sample Google web page gadgets

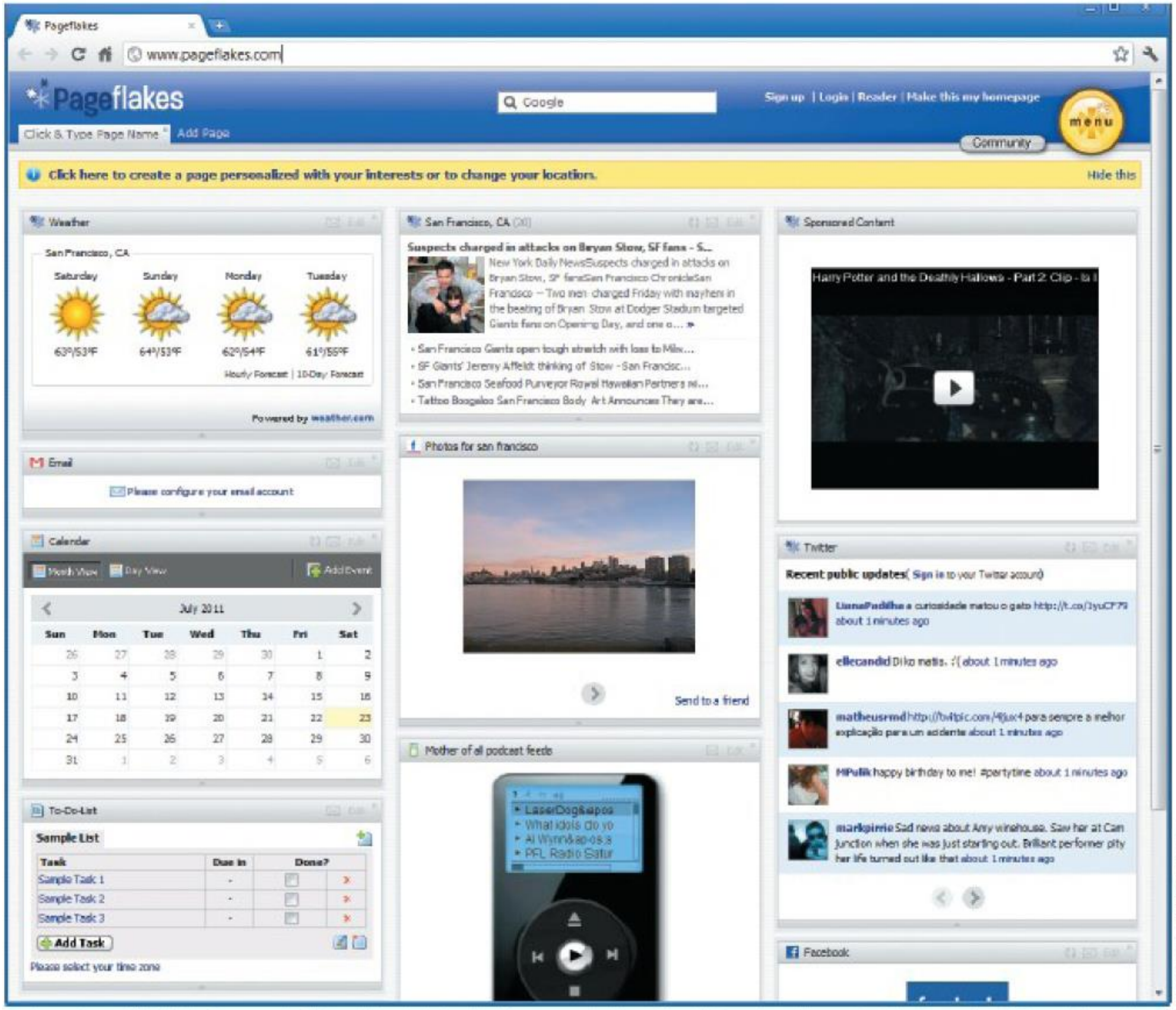


Courtesy of Google

The web as an application platform (2)

- Toolbar – a type of add-on usually comprised of iconic menu items that access the capabilities of the application or plug-ins in a user-friendly fashion
- Web mini-app – a software application that provides a complete set of functions but that must be executed within the confines of another application
- Mashup – a type of Web site that combines the functionality of several other Web sites through the use of predefined APIs
- Rich Internet Applications (RIAs) –a type of Web site that provides active user interaction as well as delivers rich multimedia

Mashup of several API to create a dashboard page



Click here to create a page personalized with your interests or to change your location.

San Francisco, CA

Saturday	Sunday	Monday	Tuesday
63°/53°F	64°/53°F	62°/54°F	61°/55°F

Hourly Forecast | 10-Day Forecast

Please configure your email account

July 2011

Sun	Mon	Tue	Wed	Thu	Fri	Sat
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

Sample List

Task	Due in	Done?	
Sample Task 1	-	□	✕
Sample Task 2	-	□	✕
Sample Task 3	-	□	✕

Please select your time zone

Suspects charged in attacks on Bryan Stow, SF fans - S...

New York Daily NewsSuspects charged in attacks on Bryan Stow, SF fansSan Francisco ChronicleSan Francisco -- Two men charged Friday with mayhem in the beating of Bryan Stow at Dodger Stadium targeted Giants fans on Opening Day, and one o...

- San Francisco Giants open tough stretch with loss to Milw...
- SF Giants' Jeremy Affeldt thinking of Stow - San Francisc...
- San Francisco Seafood Purveyor Royal Hawaiian Partners wi...
- Tattoo Boogaloo San Francisco Body Art Announces They are...

- LaserDog&apos
- What idols do yo
- Al Wynn&apos;s a
- PFL Radio Satur

Harry Potter and the Deathly Hallows - Part 2 Clip - Is I

Recent public updates(Sign in to your Twitter account)

LianaPadilha a curiosidade matou o gato http://t.co/1yuCF79 about 1 minutes ago

ellecandid Diko natis. :'(about 1 minutes ago

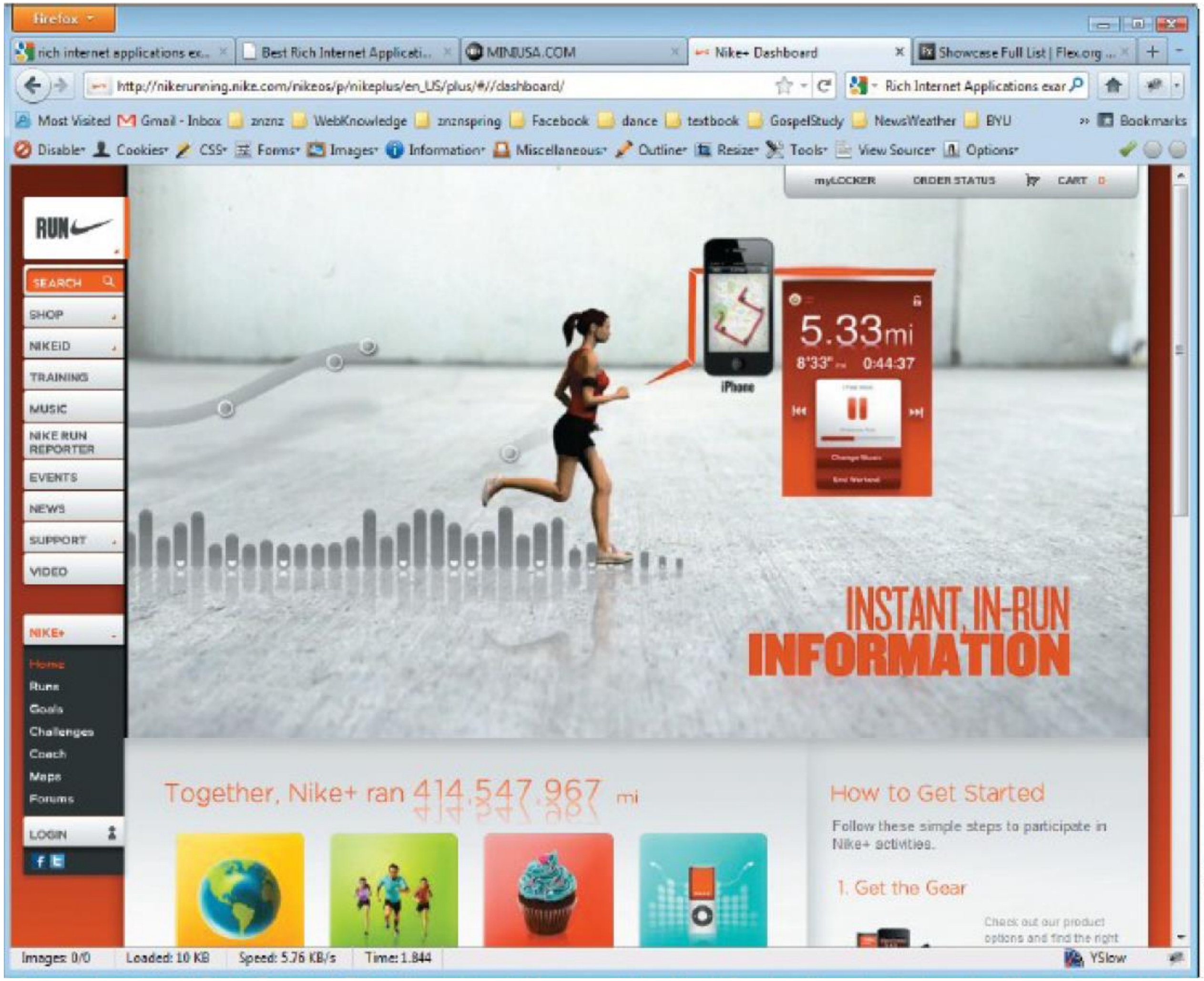
matheusrmd http://twitpic.com/4fjux4 para sempre a melhor explicação para um acidente about 1 minutes ago

MPulik happy birthday to me! #partytime about 1 minutes ago

markpirrie Sad news about Amy winehouse. Saw her at Cam Junction when she was just starting out. Brilliant performer pity her life turned out like that about 1 minutes ago

Courtesy of Pageflakes

Rich Internet Application (RIA)



Together, Nike+ ran 414,547,967 mi

How to Get Started

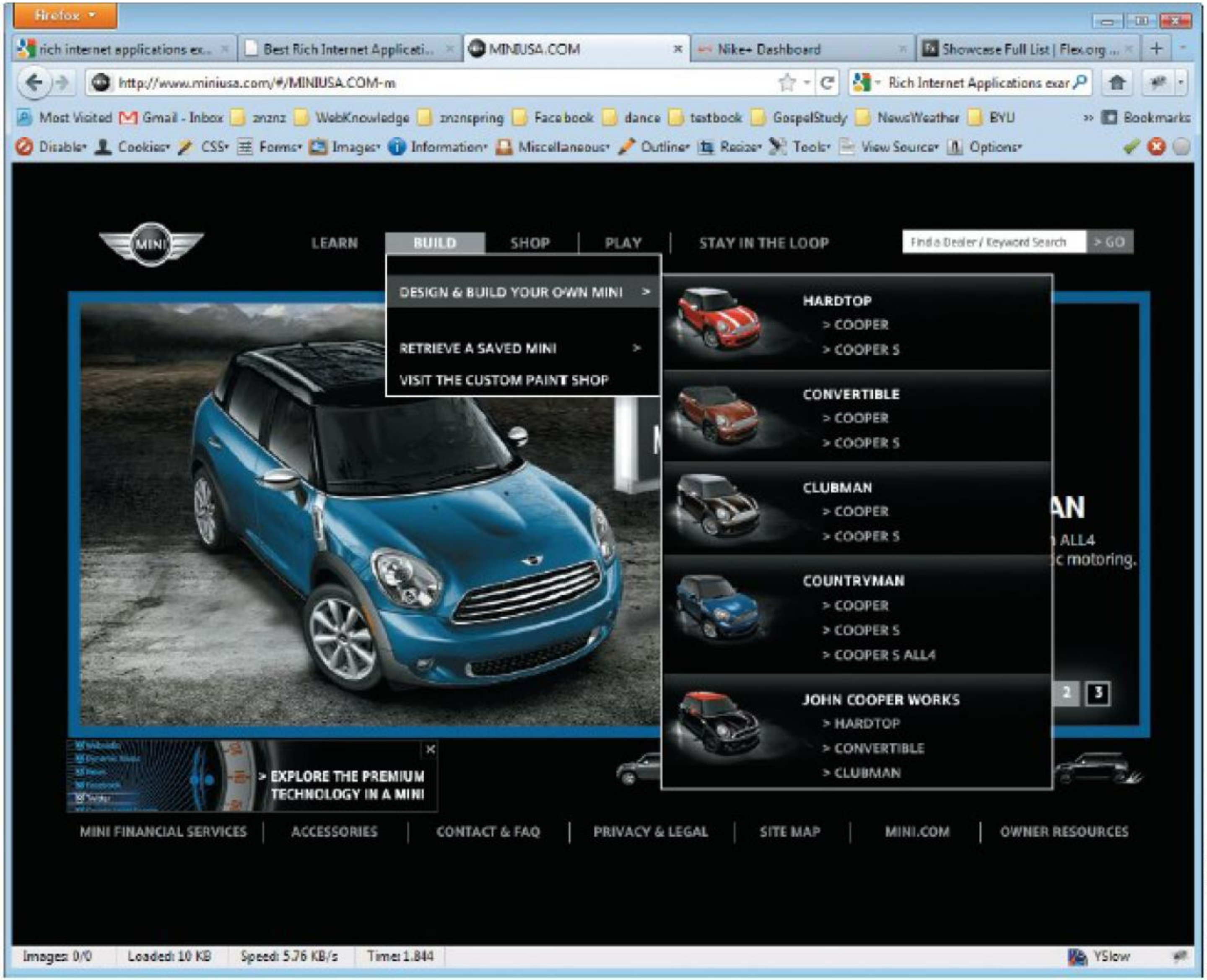
Follow these simple steps to participate in Nike+ activities.

1. Get the Gear

Check out our product options and find the right

Courtesy of nikerunning.nike.com

Rich Internet Application (RIA)



Courtesy of Miniusa.com

Rich Internet Applications (RIAs)

- JavaScript and Ajax Libraries
  - De facto standard for adding desktop-like computing within browsers
- ICEFaces and JavaFX
  - Open-source framework that provides Java language APIs to build and deploy server-based RIAs
- Adobe Flash Platform
  - Best known as a multimedia platform for animation and interactivity
- Microsoft Silverlight
  - Microsoft's Windows Communication Foundation (WCF) .NET RIA Services to support the development and deployment of RIAs
- HTML5
  - New HTML specification that standardizes RIA specifications for built-in browser delivery

Summary

- One of the most active trends in software development is adaptive development methodologies that use iterative SDLCs
- The most formal adaptive methodology is the Unified Process (UP) – one of the first to be formalized with specific definitions for iterations and processes, but it can also be agile
- Other more radical adaptive methodologies are now being promoted and used, such as Extreme Programming and Scrum
- Major trends in technology include mobile computing devices, software availability, and Web access

Summary (2)

- The number of computers will soon be surpassed by the number of mobile devices, such as smart phones and tablet computing devices
- These devices have become so pervasive because of the expanded availability of Internet access through wireless and telephone connections
- The availability of so many connected computing devices has necessitated the growth of large-scale data centers with very large server farms
- There is a trend toward sharing software applications among many users and organizations called Software as a Service (SAAS),

Summary (3)

- Another trend is providing software applications free of charge as open-source software
- A final—and extremely important—trend is the use of the Web as an application platform
- Web-based software applications provide APIs so Web-based applications can share functionality and even be combined to provide new uses of Web-based software
- Rich Internet Applications (RIAs) allow a Web-based application to function much like a desktop application, with multimedia and active user interactions

Forbs

https://www.forbes.com/sites/forbestechcouncil/2018/10/05/15-predictions-for-the-next-big-thing-in-software-development/#14e38730522f