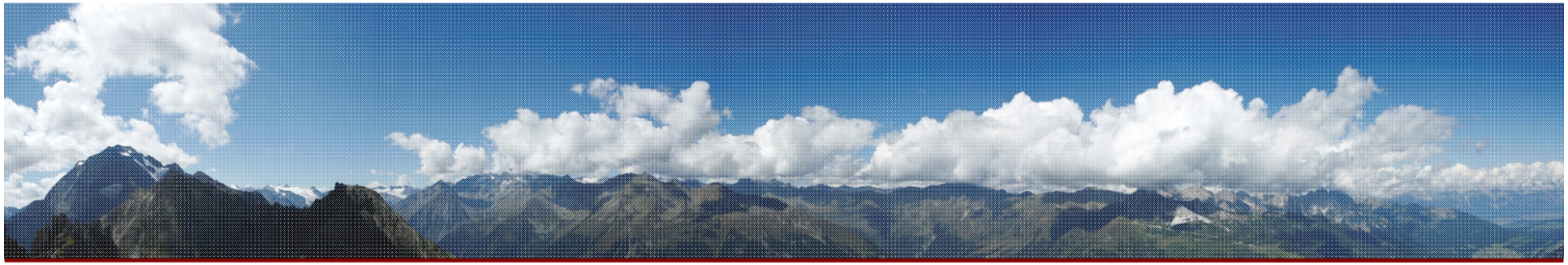
Web Services

Web Technologies



Where are we?

STI · INNSBRUCK

#	Title
1	Distributed Information Systems
2	Middleware
3	**Web Technologies**
4	Web Services
5	Basic Web Service Technologies
6	Web 2.0 Services
7	Web Service Security

Outline

STI · INNSBRUCK

- Motivation
- Technical solution
  - Exchanging Information over the Internet
  - Web Technologies for Supporting Remote Clients
  - Application Servers
  - Web Technologies for Application Integration
- Possible extensions
- Summary
- Resources

Motivation

Motivation

STI · INNSBRUCK

- Data and services often need to be shared across the boundaries of a single company or business unit:
  - Integration of different branches of the same company.
  - Automation of business processes that encompass several companies.
- The Web emerged as a technology for **sharing information** over the Internet.
  - It quickly became a medium for **connecting remote clients and servers** across the Internet.
  - More recently (with the advent of Web services) it became a medium **for integrating applications across the Internet**.
- This lesson aims at introducing basic Web technologies that are used to implement “Web” portion of Web services.

Technical Solution

Exchanging Information over the Internet

Exchanging Information over the Internet
Brief history

STI · INNSBRUCK

- In 1969 ARPA connected four universities in the US, building the network called ARPANET
  - The connected systems were autonomous and heterogeneous.
  - First standardization bodies were formed to govern the development of the network.
- One of the most prominent standards developed then is TCP/IP.
- Before the Web there were some other standards
  - Simple Mail Transfer Protocol (SMTP) – which is still the way to send e-mail
    - Later extended with Multi-purpose Internet Mail Extensions (MIME)
  - Telnet protocol
  - File Transfer Protocol (FTP)
    - Arrived soon after SMTP and Telnet.
    - Permitted anonymous file transfers.
  - Archie
    - Used FTP to create a distributed file system, index FTP archives and search through them.
  - Gopher
    - Simple client/server system and GUI for or distributing, searching, and retrieving text documents over the Internet.
- Core Web technologies (HTTP, HTML, Web servers and browsers) are evolution of those early technologies.

Exchanging Information over the Internet
Hyper Text Transfer Protocol (HTTP)

STI · INNSBRUCK

- Generic stateless protocol governing file transfer across a network.
- Originally developed by European Laboratory for Particle Physics (CERN)
  - The idea was to enable researchers to share their results and knowledge in a fast, easy and convenient manner.
- The same team came up with the name World Wide Web
  - The idea is today promoted and governed by WWW Consortium (W3C).
- Designed to support hypertext documents
  - HTTP supports Hyper Text Markup Language (HTML).
  - HTML defines standard set of markups used to render the information for human consumption.

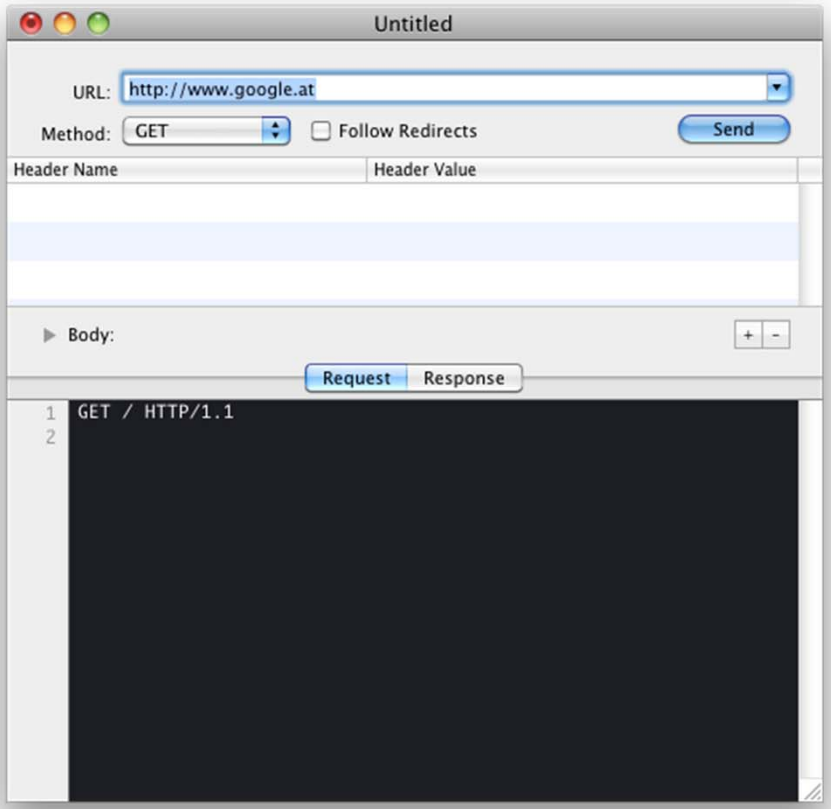
- HTTP documents are identified by Uniform Resource Identifiers (URIs).
  - URIs come in two flavors: Uniform Resource Locators (URLs) and Uniform Resource Names (URNs).
- URLs are the dominant way to identify documents over the Web.
  - In addition to identifying a resource, a URL provides a means to locate it
- A URL defines the *name of the protocol* (i.e., *scheme*) which should be used to access the document, the *address of the machine* where the resource is located, and *hierarchical description of the resource location* (and more like *query string*, and *anchor*).
- Documents can be static or dynamic
  - Dynamic documents are partially or in whole generated upon request.

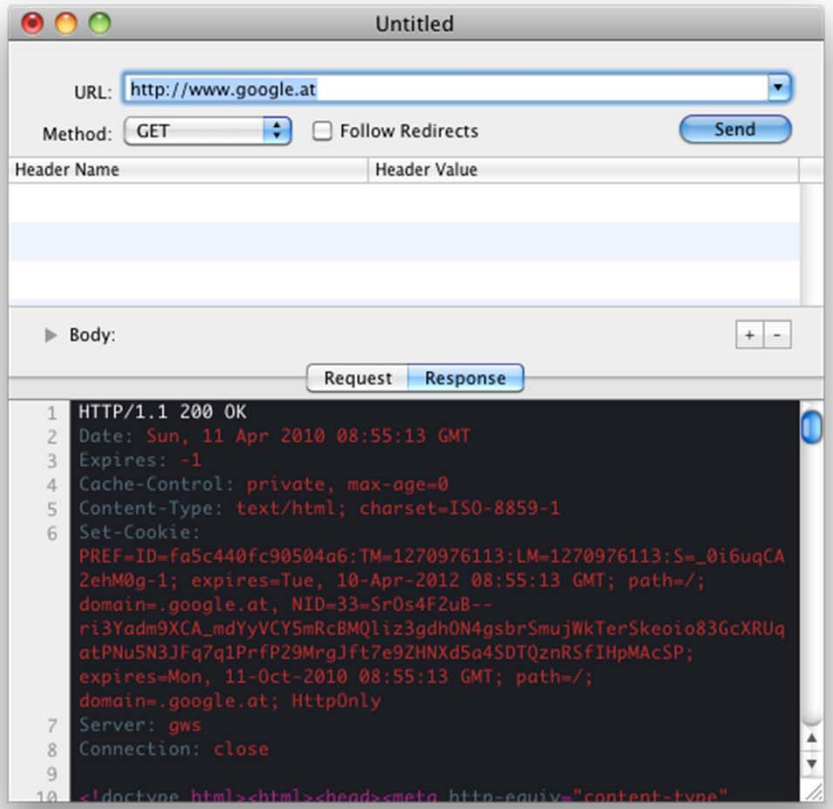
- HTTP underlying mechanism is Client/Server.
- HTTP typically relies on TCP/IP sockets.
- Starting from version 1.1 persistent connections are also supported.
- Most frequently used request methods are
  - OPTIONS – sends information about the communication options supported by a particular server,
  - GET – retrieves the specified document,
  - POST – appends or attaches the included data to the specified resource,
  - PUT – stores the included data at the location specified by the request, and
  - DELETE – deletes the resource indicated by the request.

Exchanging Information over the Internet

Hyper Text Transfer Protocol (HTTP) - Example

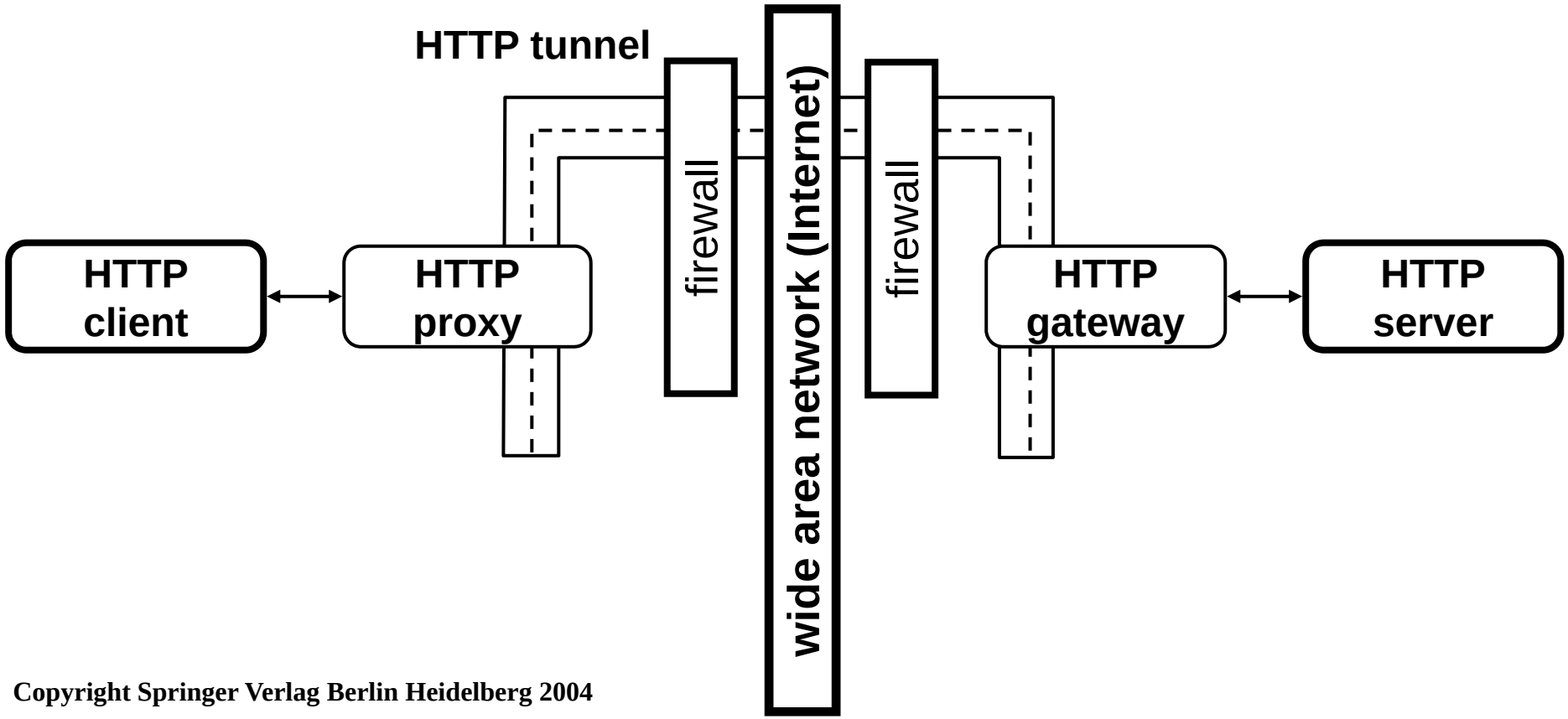
STI · INNSBRUCK





- Proxy (RFC 2616)
  - Intermediary program acting both as server and client for the purpose of making requests on behalf of other clients.
  - Potentially processes the URL and content.
- Gateway
  - Server acting as intermediary for some other server for the requested resource.
  - Acts on behalf of a server.
  - Potentially processes the URL and content.
- Tunnel
  - An intermediary program which is acting as a blind relay between two connections.
  - Used to connect two networks.
  - Doesn't process anything.

- Intermediary systems are enabling integration in the Web environment.



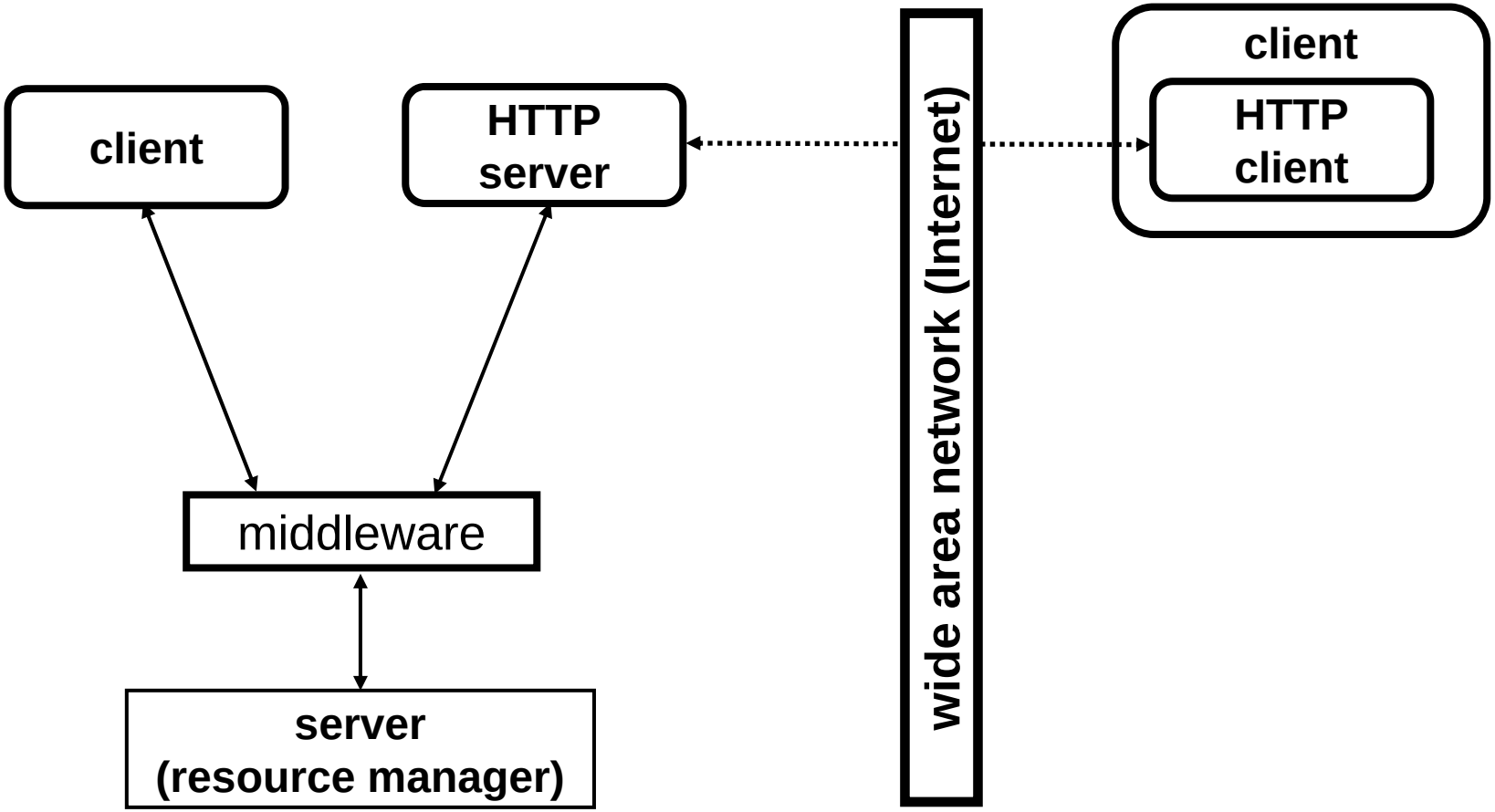
- No data encryption
  - Secure Socket Layer (SSL) developed by Netscape (1996), and its successor Transport Layer Security (TLS).
    - Relies on public key encryption to protect data transferred over TCP/IP.
  - Hyper Text Transfer Protocol over TLS/SSL (HTTPS)
    - Allows Web server and client to use TLS/SSL to authenticate to each other and establish an encrypted connection between themselves.

- Protocol is stateless
  - Information is not shared across HTTP request/response roundtrips.
  - Application developer is responsible for maintaining the relationships (i.e., state).
  - HTTP Cookies developed by Netscape (1994).
    - Enabling deployment of small data structures on the client machine on behalf of Web server.
    - They can maintain state information, can be used for personalization, tracking, session management.

Technical Solution

Web Technologies for Supporting Remote Clients

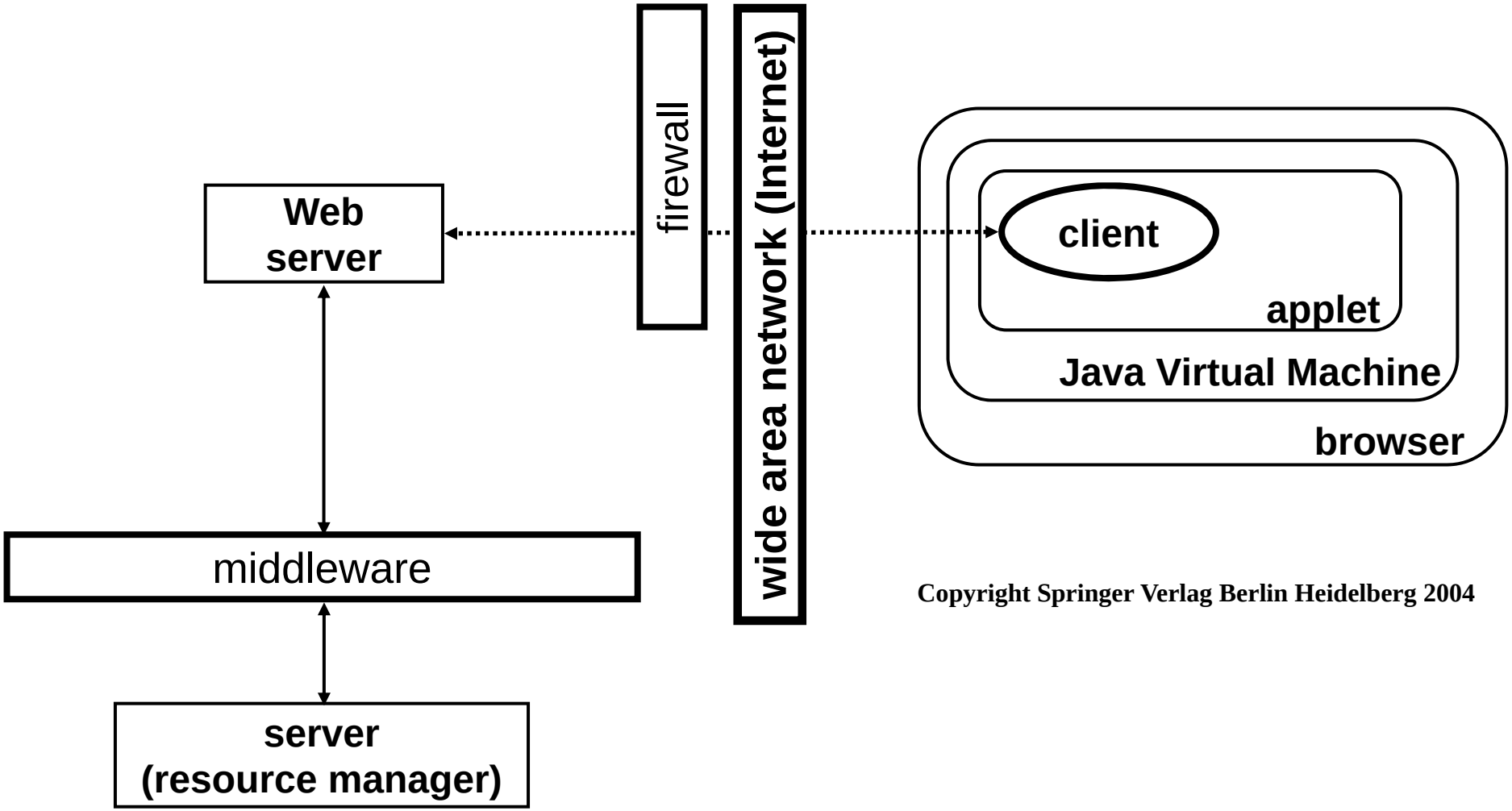
- Conventional middleware is assuming operation inside of the safe company boundaries.
- Information systems today are opening for some other users (e.g. customers)
  - Usage of Automatic Teller Machines (ATMs) by banks gives customers easier access to their accounts.
  - Manual work when dealing with customers disappears which reduces costs for banks.
- ATMs are not in a personal possession and they still incur some costs for customers (they need to travel to use the provided services)
  - Once the customer owns its personal ATMs (i.e. client) possibilities are endless – advanced applications, no usage constrains, etc.
- These are Business-To-Customer (B2C) operations
  - Customer is directly accessing company services.
  - Without Web technologies it would be quite complex to achieve efficient B2C.



- Building sophisticated applications at the client side (i.e. on Web browser side) is difficult.

- An applet is a Java program which can be embedded in an HTML document
  - Introduced by Sun Microsystems with the first version of Java language (1995).
  - The program is executed inside a Java Virtual Machine (JVM) in a controlled manner.
  - Client code (Java classes and associated artifacts) is sent to the client.
  - Applets turn the Web browser into an application-specific client without complex (re)configurations and installation procedures.
  - Applets are transient
    - Their lifetime is associated to the running browser instance.
    - Inadequate to support complex client code or frequent interactions.

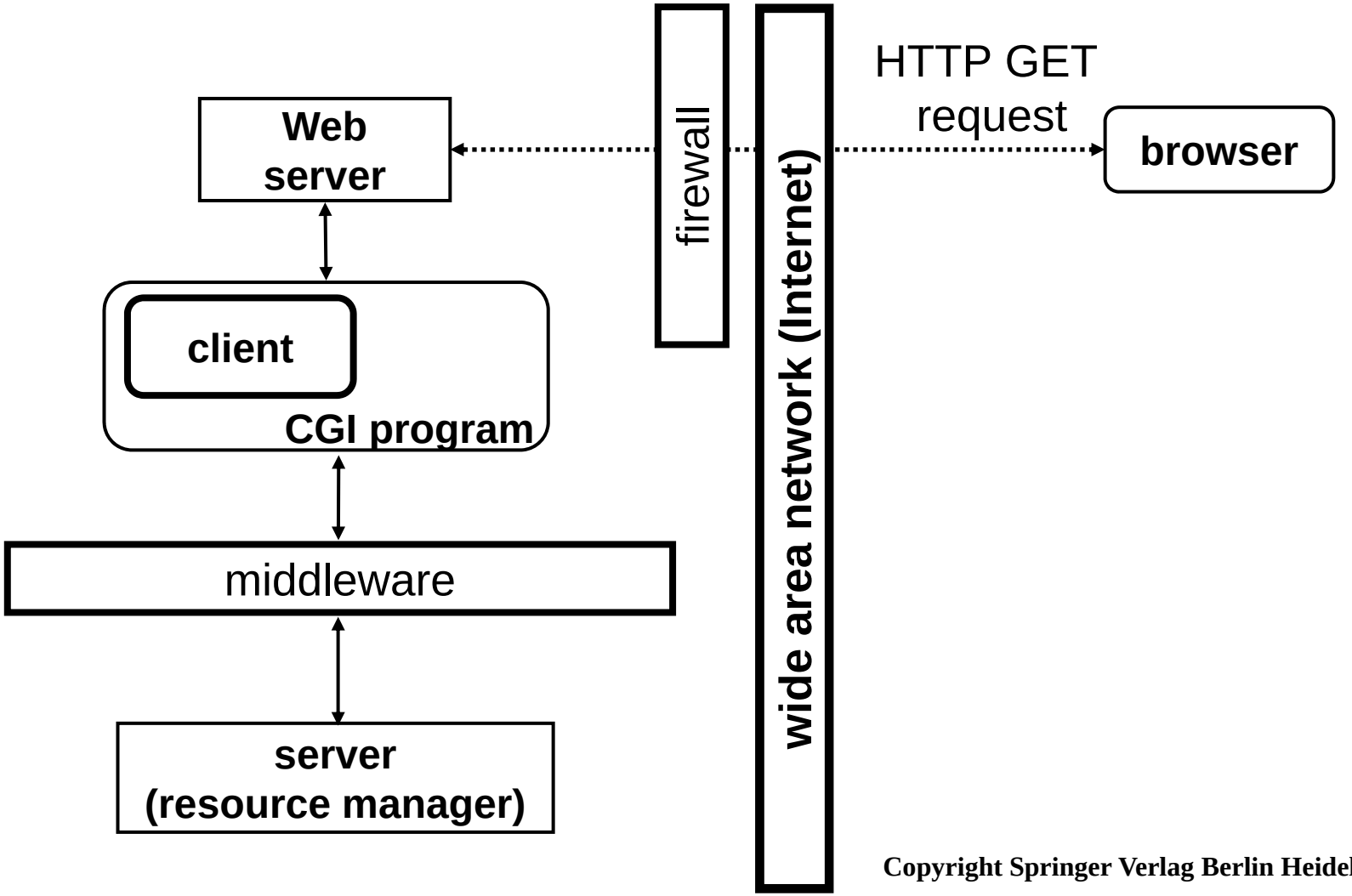
STI · INNSBRUCK



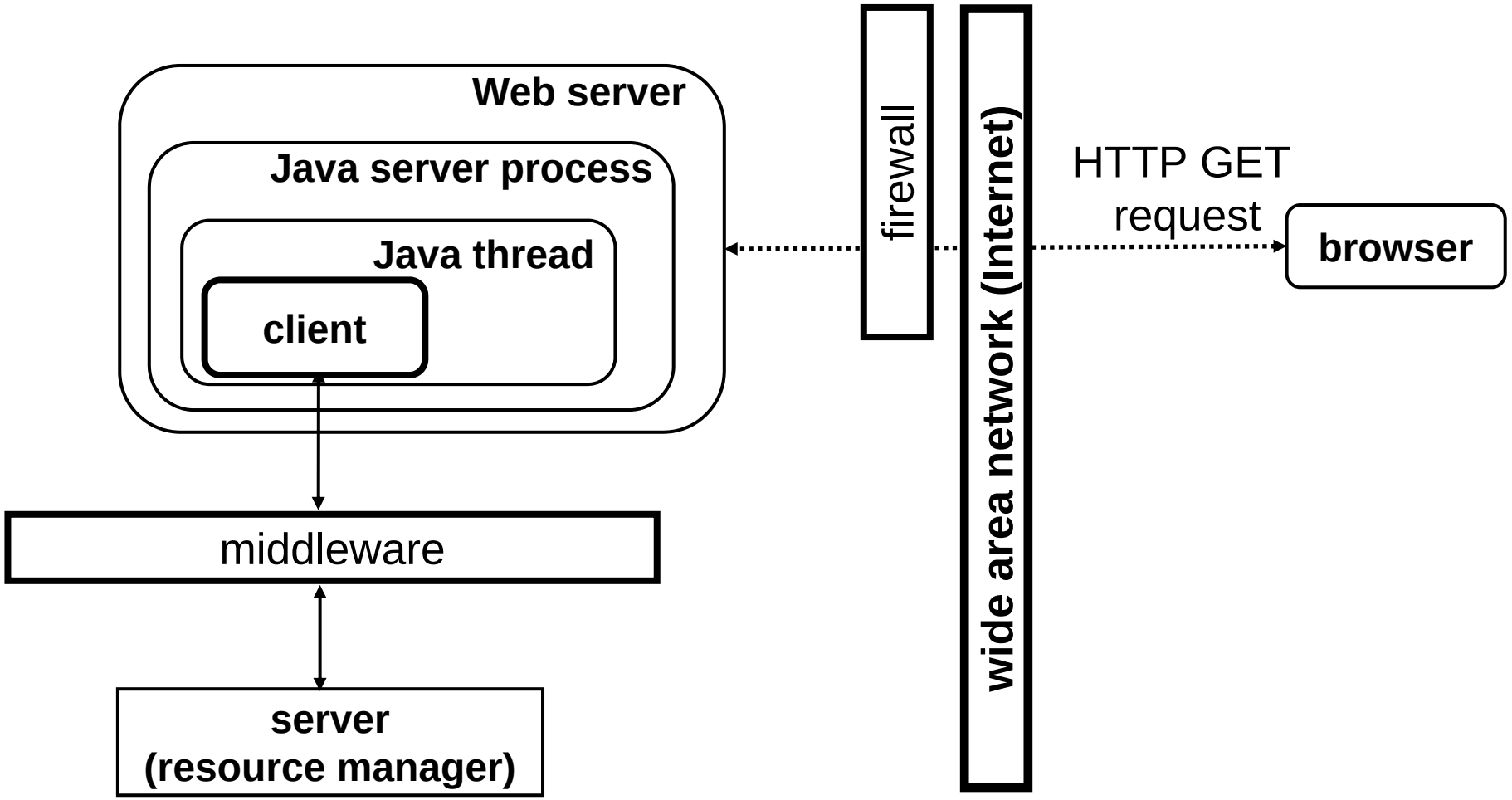
- CGI is usually used to enable server to serve content from dynamic sources (e.g. publishing information from databases).

- It basically enables HTTP server to interface with external applications (they can serve as “gateways” to the local information system).
  - CGI establishes binding between a program and a requested URL.
  - Program arguments are sent as part of requested URL.
  - CGI programs can be written in various languages.
  - A program is started as a separate process and it interacts with the underlying middleware.

STI · INNSBRUCK



- CGI programs involve certain overhead
  - Creation of separate processes limits scalability of the system.
- Java servlets use exactly the same idea as CGI scripts but
  - Servlets are running as threads of a Java server process.
  - They run as a part of the Web server.
  - They are invoked directly by embedding servlet-specific information within an HTTP request.
  - This minimizes overhead, reduces memory requirements, and enables persistent context management in contrast to CGI scripts.
    - Session tracking, sharing of DB connections, caching as optimization technique.
- The successor of servlets – Java Server Pages (JSP).



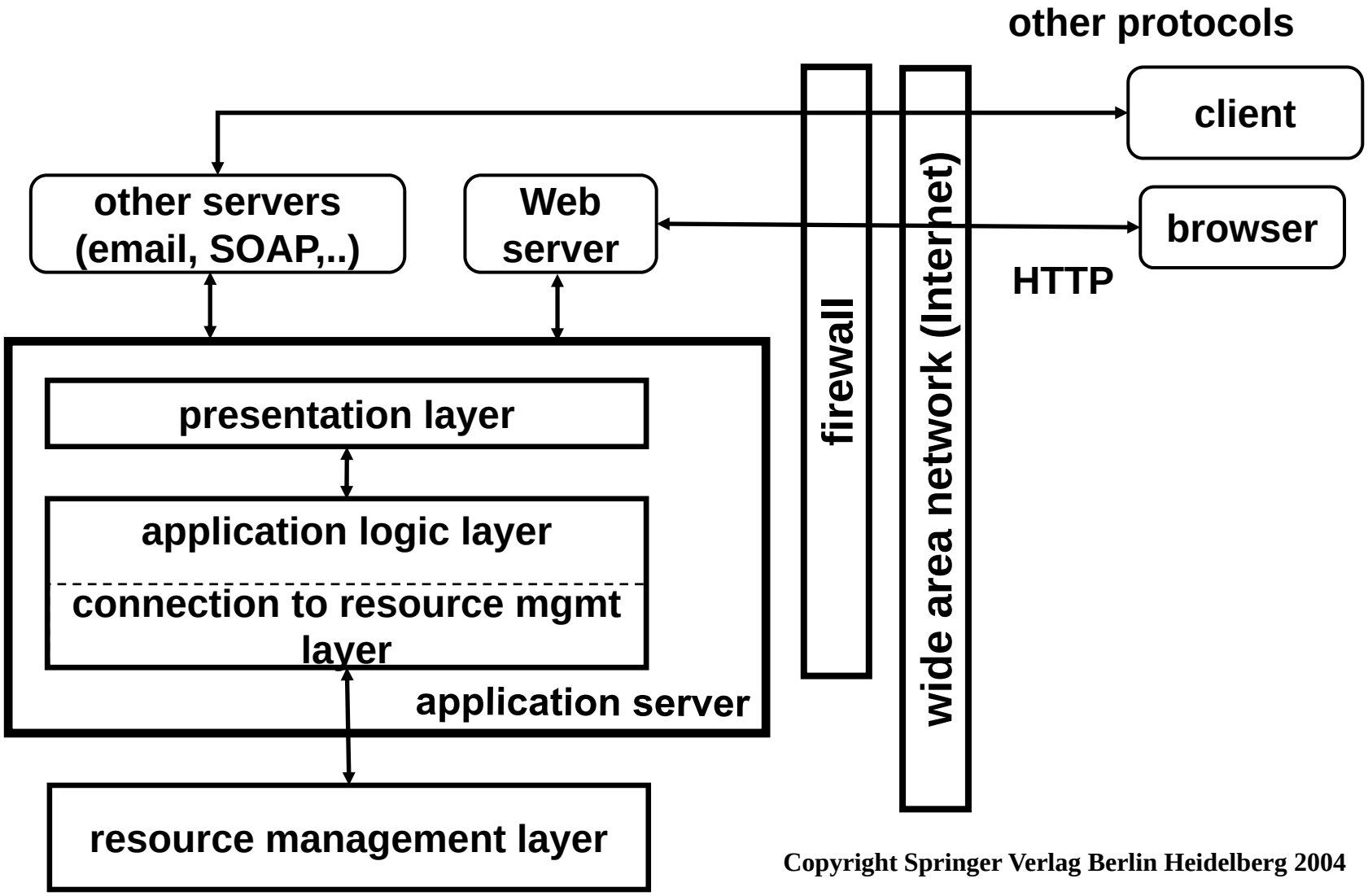
Technical Solution

Application Servers

STI · INNSBRUCK

- Increased usage of the Web as a channel to access information systems pushed middleware platforms to provide Web access support.
- Application server is a middleware system equipped with Web access channels to the services implemented using the middleware.
- The implications are following:
  - Presentation layer plays more significant role than in conventional middleware (information is exchanged through documents).
  - Presentation layer related to the Web and application layer are getting merged in order to allow an efficient delivery of content through the Web and to simplify Web application management.
- Connectivity to the resource management layer is achieved through the standardized connection architectures and APIs (JDBC, ODBC).

STI · INNSBRUCK



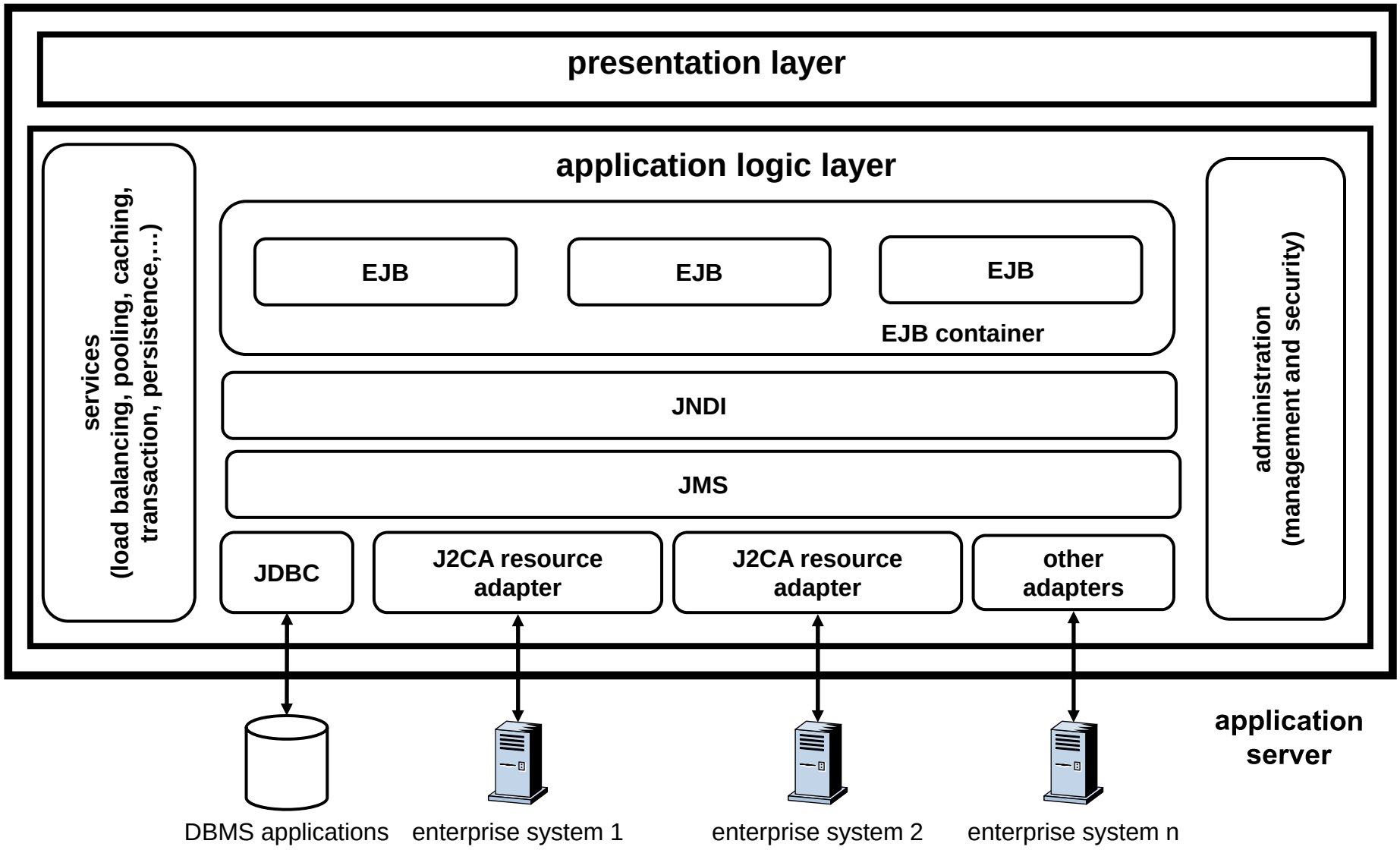
Application Servers
Support for the Application Logic

STI · INNSBRUCK

- Conceptually resemble conventional middleware platforms.
  - Functionality similar to CORBA, TP monitors, and message brokers.
- Application server vendors are targeting to produce a unique environment to host all kinds of app logic, Web-based or the other.
- The supporting services (transactions, security and persistence) are provided in an automatic and transparent manner.
- J2EE application logic support concentrates on
  - Enterprise Java Beans (EJB),
  - Java Naming and Directory Interface (JNDI), and
  - Java Message Service (JMS).
- EJB container hosts EJBs and enables application of logistic services through the mechanisms of interception.

Application Servers
Support for the Application Logic – J2EE AS architecture

STI · INNSBRUCK



Application Servers
Support for the Application Logic

STI · INNSBRUCK

- Application servers can provide services to simplify the administration and management of the applications and provide for performance and high availability
  - Caches, loading pools, monitoring, security, object administration, etc.
- Analogous features are provided by CORBA and COM+
  - These features lower the total cost of ownership (TCO).
- App servers still cannot match performance of TP Monitors.

- CGI takes black-box approach in implementing the presentation layer of a Web application.
  - Link to middleware platform without requiring changes to it.
  - It's a good solution for legacy applications.
  - True integration requires middleware cooperation (middleware should be modified to provide the necessary support to make its services accessible through the Web).
- Application servers provide more sophisticated implementations which are making the transition between Web documents and programming abstractions more efficient, flexible and manageable.
  - Variety of presentation features to support delivery of dynamically generated, personalized content (i.e., documents) to different types of clients: Web browsers, applications, devices, email-programs, Web service clients.

Application Servers
Support for the Presentation Layer

STI · INNSBRUCK


client



servers for other connections (e.g., WAP)
E-mail server
Web server
services (load balancing, pooling, caching,...)
multidevice content delivery
Servlets
JSPs
XML support
Web services support
personalization logic
administration (management and security)
presentation layer
application logic layer
connection to resource mgmt layer
resource management layer
application server

- Web browsers
  - The dynamic components which generate HTML are integrated with the application logic layer.
  - JSP fragments can be easily linked with EJBs through EJB containers.
  - Applets may interact with app server through RMI and CORBA/IIOP.
- Devices
  - Smaller devices may use XHTML Mobile Profile.
  - Dynamic generation of content in different markup languages and automatic conversion between these languages is also needed.
  - Devices may be peculiar and this may impose some problems in terms of information organization, etc. It may require different presentation (and even application) logic to be written.
- Email-programs
  - App server may support packaging of information which is delivered through SMTP.
- Web service clients
  - Require support for SOAP.
- App servers may also offer personalization services
  - Supported through the definition of condition/action rules.

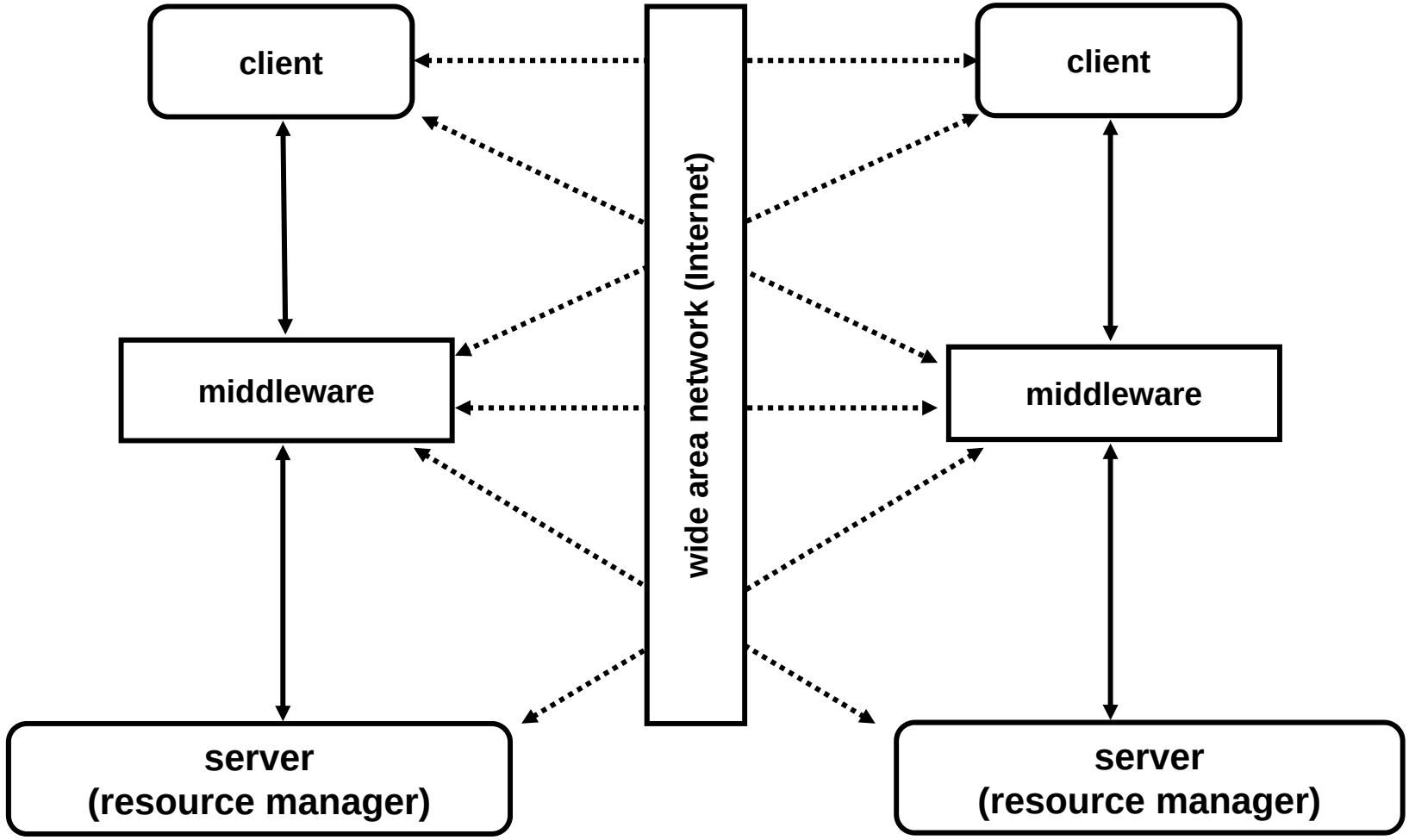
Technical Solution

Web Technologies for Application Integration

STI · INNSBRUCK

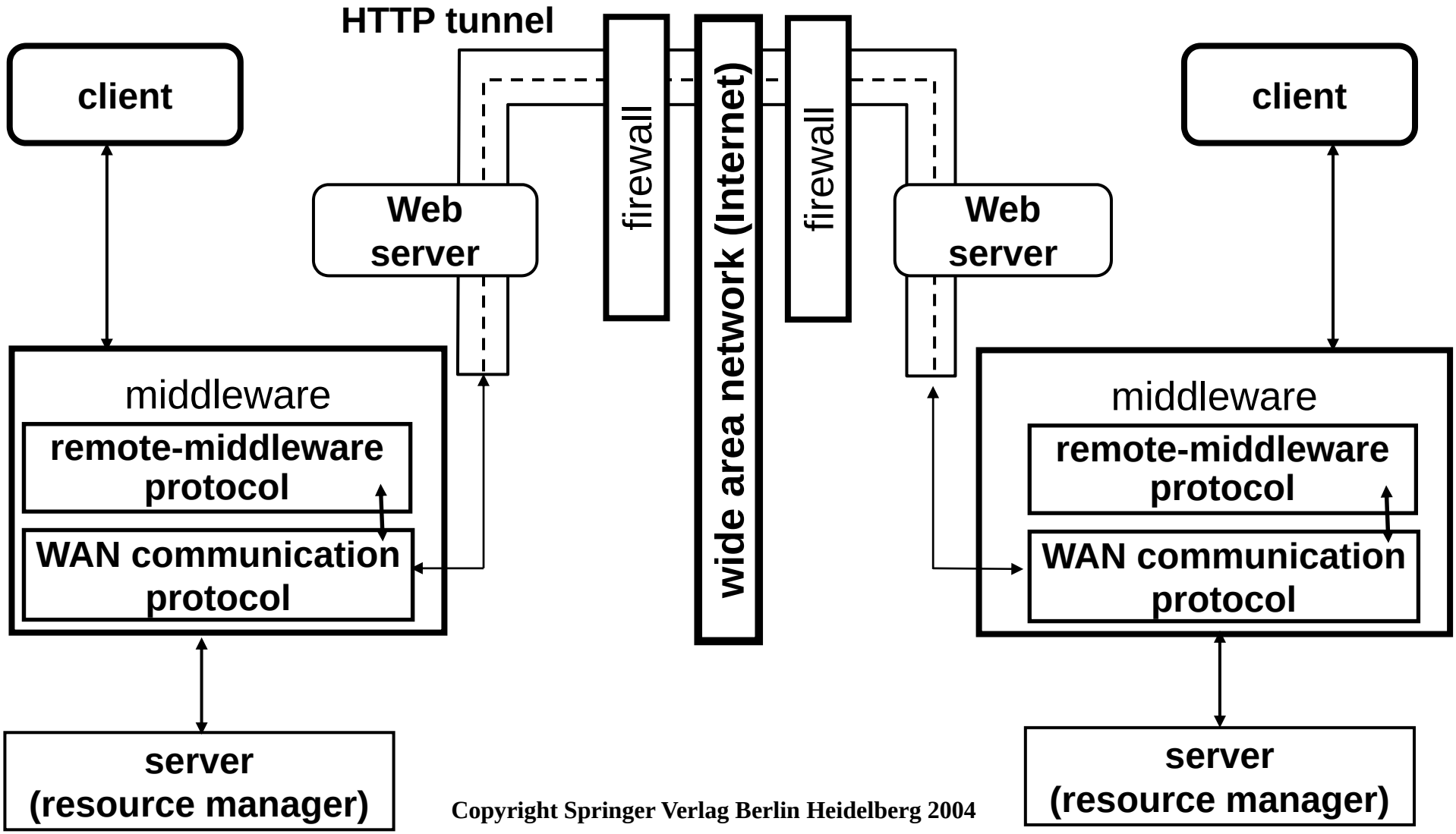
- Integrating systems that are separated by a wide area network.
- Almost all combinations between the constituent parts of a 3-tier architecture are possible.
- The suitable strategy is chosen on top of a number of factors such as
  - level of standardization,
  - communication protocols, etc.
- Before the pervasive Web only two of the strategies were realistic:
  - specialized clients at the level of message exchanges, and
  - direct integration of middleware layers (if they are compatible).

STI · INNSBRUCK



- Conventional middleware is extended to support the Internet as one of the access channels.
- Different branches of the same company may implement their own middleware solutions which eventually must communicate to each other.
- Business-To-Business (B2B) electronic transaction exchanges
  - Enable inter-company communication.
  - Faster, more cost effective, less error prone, logs data for monitoring and analysis.
- Services must be invoked which reside in the different company branches.
  - Relatively easy to do – CORBA allows this through Internet-wide integration of ORBs by relying on the General Inter-ORB Protocol (GIOP).
  - Call over GIOP is translated to IIOP and then finally to TCP/IP call.
  - But ORBs are typically hidden behind firewalls, problem of agreeing upon interface definitions and data formats exists, directory service is needed to discover objects.

- Firewall is a barrier against unwanted network traffic
  - It blocks many communication channels in doing so (almost all forms of communication employed by conventional EAI products cannot traverse a well-configured firewall - no usage of RCP, RMI, GIOP/IIOP).
- In traditional middleware this problem never appears.
  - Every component in the system trusts and knows every other component.
- The solution is to trick the firewall into believing that traffic is actually allowed
  - Tunneling stands for packaging the hidden protocol into the allowed one.
  - It is common to enable tunneling through HTTP or SSH.
  - Intermediary must be used to convert original message into HTML or XML, send document using HTTP, extract the message from one document once it reaches the recipient.
  - Resulting architecture is rather cumbersome as it requires yet additional layer and introduces another level of indirection.



- Electronic Data Interchange For Administration, Commerce and Transport (EDIFACT) is developed under UN and standardized in ISO 9735.
- Identifying common syntax and semantics for the exchanged data between companies is of primary interest
  - Interpretation of received data must be unambiguous.
- In conventional middleware the problem is hidden behind IDLs
  - intermediate data representation helps bridging the difference between operating systems and computer architectures.
- In message-oriented system they are determined by EDIFACT standard
  - Provides standard templates for the messages and for the content of the messages.
  - A message is encoded in ASCII and handled by converters.
  - EDIFACT defines an extensive collection of standardized message types: Quality data, Purchase Orders, etc.
- EDIFACT facilitates commercial electronic exchanges
  - Parsers can be generated to automatically take data from a message and deliver it to the application logic.
- The downside is that EDIFACT wants to do much, much more that it is meant – it's a very complex standard.

- EDIFACT can be used for exchanges that have been standardized.
  - Information exchanged over the Web is much more varied and richer for any such a standardization attempt.
- XML focuses on syntax, rather than semantics of the documents exchanged.
  - Clearly defined structure helps users to understand semantics of the different parts of the document.
- Tools for parsing and extracting info are standardized.
  - Validation is supported through XML Schema and DTD.
- In contrast to EDIFACT XML program needs additional understanding of how to process the information.
  - Standardization bodies can specify XML document types, and define meaning (often in plain English) which can be exchanged by parties – e.g. RosettaNet.

Possible Extensions

Possible Extensions

- The Semantic Web
  - An extension and evolving development of the World Wide Web.
  - Semantics of information and services is precisely defined.
  - The data can be “understood” in the process of satisfying requests of people and machines while processing the Web content.
  - The Semantic Web specifications and languages (RDF(S), OWL, RIF, etc.) are built on top of the Web specifications and languages.

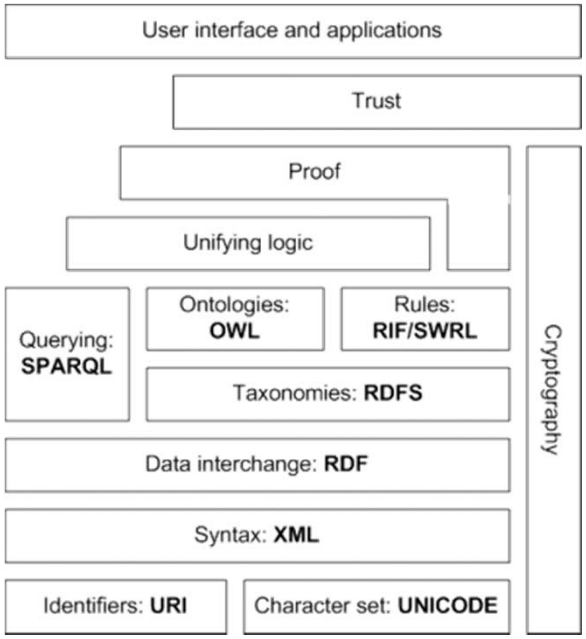


Figure taken from http://en.wikipedia.org/wiki/Semantic_web

Summary

Summary

STI · INNSBRUCK

- Web technologies enable data and services sharing across the boundary of a single company
- In a nutshell core Web technologies (HTTP, HTML, URI) enable document-style information exchange over the Internet
- Various solutions have been introduced on top of the basic Web technologies in order to
  - Support remote clients (Java Servlets and Applets, CGI scripts),
  - Extend middleware towards Web (Application Servers),
  - Enable Internet-wide application integration (middleware extensions, HTTP tunneling, common data representation formats).

References

References

STI · INNSBRUCK

- Mandatory reading
  - Gustavo Alonso, Fabio Casati, Harumi Kuno, and Vijay Machiraju. Web Services - Concepts, Architectures and Applications. Springer-Verlag, 2004. **4th chapter**

- Wiki and Web reference
  - ARPANET http://en.wikipedia.org/wiki/ARPANET
  - TCP/IP http://en.wikipedia.org/wiki/TCP/IP
  - SMTP http://en.wikipedia.org/wiki/SMTP
  - MIME http://en.wikipedia.org/wiki/MIME
  - Telnet http://en.wikipedia.org/wiki/Telnet
  - FTP http://en.wikipedia.org/wiki/FTP
  - Archie search engine http://en.wikipedia.org/wiki/Archie_search_engine
  - Gopher http://en.wikipedia.org/wiki/Gopher_(protocol)
  - HTTP http://en.wikipedia.org/wiki/HTTP
  - HTML http://en.wikipedia.org/wiki/HTML
  - URI http://en.wikipedia.org/wiki/URI
  - URL http://en.wikipedia.org/wiki/URL

References

STI · INNSBRUCK

- Wiki and Web reference (cont'd)
  - TLS/SSL http://en.wikipedia.org/wiki/Transport_Layer_Security
  - HTTPS http://en.wikipedia.org/wiki/HTTPS
  - B2C http://en.wikipedia.org/wiki/B2C
  - CGI http://en.wikipedia.org/wiki/Common_Gateway_Interface
  - JSP http://en.wikipedia.org/wiki/JavaServer_Pages
  - Java Servlet http://en.wikipedia.org/wiki/Java_Servlet
  - Java Applet http://en.wikipedia.org/wiki/Java_Applet
  - J2EE http://en.wikipedia.org/wiki/J2EE
  - EJB http://en.wikipedia.org/wiki/EJB
  - JNDI http://en.wikipedia.org/wiki/JNDI
  - JMS http://en.wikipedia.org/wiki/Java_Message_Service
  - XHTML Mobile Profile http://en.wikipedia.org/wiki/XHTML_Mobile_Profile
  - B2B http://en.wikipedia.org/wiki/Business-to-business
  - EDIFACT http://en.wikipedia.org/wiki/EDIFACT
  - XML http://en.wikipedia.org/wiki/XML
  - RosettaNet http://en.wikipedia.org/wiki/RosettaNet

Next Lecture

STI · INNSBRUCK

#	Title
1	Distributed Information Systems
2	Middleware
3	Web Technologies
4	**Web Services**
5	Basic Web Service Technologies
6	Web Service Composition
7	Web 2.0 Services
8	Web Service Security

Questions?

STI · INNSBRUCK