Knowledge Media Institute Open University, UK

Inter-Widget Communication

Lecture series: Mash-up Personal Learning Environments

Ivan Zuzak University of Zagreb, Croatia



Locate Melsettings toggle

er: Server URL

Lecture Outline

- Motivation
- Inter-Widget Communication and Web Browsers
- Dimensions of Inter-Widget Communication
- Example Systems
- Requirements Case Study
- Future Challenges and Conclusion

- Personal Learning Environments (PLEs)
 - Learning is dynamic and personalized
 - Flexible
 - Choice of learning elements
 - Interaction of learning elements



- Mash-up PLEs
 - Web widgets as learning elements
 - Mash-ups as the methodology
 - Learning process = coordination of widget activities
 - Widget-to-widget communication
 - Automated execution of learning process
 - Social collaboration



- Remember *Operating systems 101*?
 - Inter-process communication
 - Sockets, named pipes, ...
- Inter-widget communication is similar
 - Higher level of abstraction
 - "The browser is the new OS"

Inter-Widget Communication and Web Browsers

Inter-Widget Communication

- What is a widget?
 - Intuitive definition
 - Self-contained functionality...
 - ... accessible through a small GUI ...
 - ... embedded in a Web application
 - A "mini Web application"



• Operational, software engineering definition?





Web browser













// HTML source for Web application A
// loaded from http://urlA.com/

<html>

<head> ... </head>

<body>

<div> Container content </div>
<iframe src="http://wikipedia.com/" />
<iframe src="http://google.com/" />
<iframe src="http://kmi.open.ac.uk/" />
</body>

</html>

- Web Application
 - Tree hierarchy of window contexts
 - Top-level contexts Browser windows/tabs
 - Nested contexts Iframes



- Web Application
 - Tree hierarchy of window contexts
 - Top-level contexts Browser windows/tabs
 - Nested contexts Iframes
- Window contexts
 - HTML rendering and JavaScript execution
 - Origin-based isolation (Same origin policy)
 - http://wikipedia.com/widget
 - http://kmi.open.ac.uk/widget
 - Cross-origin interaction is restricted



- Two approaches for implementing widgets and IWC
- Iframe contexts (more common)
 - Widget
 - Iframe context, temporary entity in a Web browser
 - Widget specification
 - HTML+CSS+JS resources stored on a Web server
 - Inter-widget communication
 - Data transfer between two contexts

- Two approaches for implementing widgets and IWC
- Iframe contexts (more common)
 - Widget
 - Iframe context, temporary entity in a Web browser
 - Widget specification
 - HTML+CSS+JS resources stored on a Web server
 - Inter-widget communication
 - Data transfer between two contexts
- Virtual contexts (recent research)

Dimensions of Inter-Widget Communication

- Inter-widget communication system
 - Part of a MUPPLE which supports IWC
 - Browser primitives, JavaScript libraries

- Inter-widget communication **system**
 - Part of a MUPPLE which supports IWC
 - Browser primitives, JavaScript libraries
- Common questions
 - How to express requirements for IWC?
 - What are each IWC system's properties?
 - How to choose an IWC system?

- Software architecture approach
- System design space

- Software architecture approach
- System design space
 - Multidimensional space



- Software architecture approach
- System design space
 - Multidimensional space
 - Dimensions describe system characteristics
 - Dimension values correspond to alternatives



- Software architecture approach
- System design space
 - Multidimensional space
 - Dimensions describe system characteristics
 - Dimension values correspond to alternatives
 - A specific system corresponds to a point in the design space



I. Zuzak - MUPPLE lecture series: Inter-widget communication

- 1) Examine existing IWC systems
- 2) Define IWC design space



- 1) Examine existing IWC systems
- 2) Define IWC design space
- 3) Define MUPPLE requirements for IWC (dimension bounds)



- 1) Examine existing IWC systems
- 2) Define IWC design space
- 3) Define MUPPLE requirements for IWC (dimension bounds)
- 4) Obtain acceptable systems



I. Zuzak - MUPPLE lecture series: Inter-widget communication

- 1) Examine existing IWC systems
- 2) Define IWC design space
- 3) Define MUPPLE requirements for IWC (dimension bounds)
- 4) Obtain acceptable systems
 - 100% match choose one acceptable system
 - <100% match extend some existing system



I. Zuzak - MUPPLE lecture series: Inter-widget communication












• Type of IWC system (IWC system span)

Web browser

Web server

• Type of IWC system (IWC system span)



• Type of IWC system (IWC system span)



Type of IWC system (IWC system span)



Type of IWC system (IWC system span)



Communication model

Communication model



Communication model



- Naming
 - How are widgets addressed when using the IWC system?

- Naming
 - How are widgets addressed when using the IWC system?
 - JavaScript object references to contexts
 - A reference to the window context, with an API
 - Contexts have references to parent and nested contexts
 - window.parent, window.frames

- Naming
 - How are widgets addressed when using the IWC system?
 - JavaScript object references to contexts
 - A reference to the window context, with an API
 - Contexts have references to parent and nested contexts
 - window.parent, window.frames
 - Custom strings
 - E.g. "map coordinates" for publish-subscribe channel names, shared memory location or remote procedure name
 - Resolved internally by IWC system

- Discovery support
 - Widget names may not be known in advance
 - Widgets may be added/removed at runtime



- Discovery support
 - Widget names may not be known in advance
 - Widgets may be added/removed at runtime
 - No support
 - Supported
 - Traversal of Web application context hierarchy
 - Hidden communication between contexts
 - RPC and pubsub channel discovery



- Reliability support
 - Widgets may be added/removed/changed at runtime
 - Receiver may not be fully loaded yet
 - Procedures (RPC) or channels (pubsub) may be added dynamically

- Reliability support
 - Widgets may be added/removed/changed at runtime
 - Receiver may not be fully loaded yet
 - Procedures (RPC) or channels (pubsub) may be added dynamically
 - No support
 - Handshake + queuing
 - Widget-level handshake
 - Message-level handshake

Example Inter-Widget Communication Systems

Example IWC Systems new systems Number of Unstandardized Healthy IWC HTML5 and insecure standardization ecosystem browser hacks Time

- Today's IWC ecosystem
 - More than 30 browser primitives and client-side frameworks
 - Many more server-side frameworks
 - Twitter or Facebook messages for IWC? Why not?

- HTML5 postMessage API
 - Standardized browser primitive

Browser		
http://sender.origin.com/widget	http://receiver.origin.com/widget	
Sender widget code	Receiver widget code	

- HTML5 postMessage API
 - Standardized browser primitive



- HTML5 postMessage API
 - Standardized browser primitive



- HTML5 postMessage API
 - Communication model Message-oriented system
 - String messages
 - Naming Destination context object
 - Cross-origin support
 - Secure Origin-based authorization
 - No discovery support
 - No reliability support

- jsChannel
 - Client-side framework Built on top of postMessage

Browser		
http://sender.origin.com/widget		http://receiver.origin.com/widget
Sender widget code		Receiver widget code

- jsChannel
 - Client-side framework Built on top of postMessage

Browser		
http://sender.origin.com/widget	http://receiver.origin.com/widget	
Sender widget code	<pre>Receiver widget code jsChannel.attachRPCHandler("refreshMap", origin, handler); function handler(parameters) { // use parameters; return result; }</pre>	

- jsChannel
 - Client-side framework Built on top of postMessage



- jsChannel
 - Communication model Remote procedure call
 - Procedures, procedure call parameters
 - Naming Destination context object, procedure name
 - Secure Origin-based authorization
 - No discovery support
 - Reliability support
 - Widget-level handshake

- Faye Messaging server and browser library
 - Requires a server component for message routing



- Faye Messaging server and browser library
 - Requires a server component for message routing



- Faye Messaging server and browser library
 - Requires a server component for message routing



- Faye
 - Server-mediated communication framework
 - Ajax for browser-server communication
 - Communication model Publish-subscribe
 - String messages on channels
 - Naming String channel name
 - Cross-origin support
 - Security
 - Password protected channels
 - HTTPS communication with server
 - No discovery support
 - No reliability support

- Analysis results of over 30 IWC systems
 - Browser primitives, client-side and server-mediated coordination frameworks

- Analysis results of over 30 IWC systems
 - Browser primitives, client-side and server-mediated coordination frameworks



Number of communication models supported by systems



Analysis results of over 30 IWC systems



Requirements Case Study for Inter-Widget Communication

IWC Requirements Case Study

- Geppeto
 - Consumer programming environment
 - User-defined application logic based on GUI actions
- Geppeto
 - Consumer programming environment
 - User-defined application logic based on GUI actions



TouchMe

- Geppeto
 - Consumer programming environment
 - User-defined application logic based on GUI actions



estapp settings to	oggle	8
Geppeto	Touch	Me
Tab1 +		
Get Picture!	Add Remove	
	When clicked When double clicked When checked When unchecked When selected	
	Type in Click Double click Check Uncheck Select	
	Copy Paste	

- Geppeto
 - Consumer programming environment
 - User-defined application logic based on GUI actions





- Geppeto
 - Spreadsheet view of user-defined logic

wait for click Planvoznje1 at ivan2	wait for click Osvjezi at ivan2	
text@ivan2 => text@ZET-Planvoznje	click Osvjezi at GPSpolozaj	
click Osvjezi at GPSpolozaj	copy element3 at GPSpolozaj to element3 at ivan2	
gpsaddress@GPSpolozaj => text1@ZET-Planvoznje		
click Planvoznje@ZET-Planvoznje		
element2@ZET-Planvoznje => element2@ivan2		
copy element3 at GPSpolozaj to element3 at ivan2		

• Geppeto – IWC requirements

- Geppeto IWC requirements
 - *Type of system* Browser primitive or client-side framework

- Geppeto IWC requirements
 - *Type of system* Browser primitive or client-side framework
 - *Communication model* Remote procedure call
 - *Distribution scheme* 1:1 unicast

- Geppeto IWC requirements
 - *Type of system* Browser primitive or client-side framework
 - *Communication model* Remote procedure call
 - *Distribution scheme* 1:1 unicast
 - Maximum message length Unlimited

- Geppeto IWC requirements
 - *Type of system* Browser primitive or client-side framework
 - *Communication model* Remote procedure call
 - *Distribution scheme* 1:1 unicast
 - *Maximum message length* Unlimited
 - Cross-origin communication Yes

- Geppeto IWC requirements
 - *Type of system* Browser primitive or client-side framework
 - *Communication model* Remote procedure call
 - *Distribution scheme* 1:1 unicast
 - *Maximum message length* Unlimited
 - Cross-origin communication Yes
 - Naming Visible widget names (string) + procedure names (string)
 - Translation from widget names to context references

- Geppeto IWC requirements
 - *Type of system* Browser primitive or client-side framework
 - *Communication model* Remote procedure call
 - *Distribution scheme* 1:1 unicast
 - *Maximum message length* Unlimited
 - Cross-origin communication Yes
 - Naming Visible widget names (string) + procedure names (string)
 - Translation from widget names to context references
 - *Discovery* No preference
 - Reliability Yes

- Geppeto IWC requirements
 - *Type of system* Browser primitive or client-side framework
 - *Communication model* Remote procedure call
 - *Distribution scheme* 1:1 unicast
 - Maximum message length Unlimited
 - Cross-origin communication Yes
 - Naming Visible widget names (string) + procedure names (string)
 - Translation from widget names to context references
 - *Discovery* No preference
 - Reliability Yes
 - Authorization of sender and receiver Yes

Future Challenges and Conclusion

Virtual Contexts

- Driven by security concerns
 - Google Caja, ECMAScript5 strict mode

Virtual Contexts

- Driven by security concerns
 - Google Caja, ECMAScript5 strict mode
- Virtual context
 - Sanitized, safe subset of context content (HTML, CSS and JavaScript)
 - Embedded within host window context

Host window context - browser tab
Virtual context Widget A
Virtual context Widget B

Virtual Contexts

- Driven by security concerns
 - Google Caja, ECMAScript5 strict mode
- Virtual context
 - Sanitized, safe subset of context content (HTML, CSS and JavaScript)
 - Embedded within host window context
 - Interaction with environment possible only if given references by host context
 - IWC = method invocation



User Involvement in IWC

- Degree of user intervention required for IWC
 - a) Integrated into widgets, no user involvement required
 - b) Requires some degree of user involvement

User Involvement in IWC

- Degree of user intervention required for IWC
 - a) Integrated into widgets, no user involvement required
 - b) Requires some degree of user involvement
 - Drag-and-drop of elements
 - Wiring of elements (e.g. Yahoo Pipes)
 - Visual proximity of widgets
 - User-defined GUI-level scripts (e.g. Geppeto)

Conclusion

- Inter-widget communication in MUPPLEs
 - Tied to Web application architecture
 - Many dimensions for IWC system classification
 - Large ecosystem of existing systems
- Future research
 - Security and performance dimensions
 - User experience dimensions
 - Semantic interoperability

Resources

Virtual contexts

- http://code.google.com/p/google-caja/
- http://en.wikipedia.org/wiki/Caja_programming_language
- HTML5 Web Messaging (postMessage API)
 - http://www.w3.org/TR/webmessaging/
- Software architecture design space
 - Shaw, Garlan: "Software Architecture: Perspectives on an Emerging Discipline", Prentice Hall, 1996
- Usability
 - Isaksson, Palmer: "Usability and Inter-widget Communication in PLE", MUPPLE'10
- List of existing IWC systems and systematization
 - http://code.google.com/p/pmrpc/wiki/IWCProjects

May 3 2011

Questions?

Do we need more powerful IWC systems? What about peer-to-peer inter-browser IWC communication? But, I've heard that there are "desktop widgets" and "web widgets"? Is IWC being standardized?

Thank you!

Ivan Zuzak izuzak@gmail.com @izuzak



I. Zuzak - MUPPLE lecture series: Inter-widget communication