

Work with InterSystems.

Not separate systems.



Java and .NET

Bill McCormick

INTERSYSTEMS



- Cache 3.1 ships with our first try at a Java Projection
 - Following our typical practice of letting people choose their preferred tool to connect to Cache via Cache Object technology
 - Cache 4.1 the Java projection achieves “release” quality
- Cache 5.0 we introduce the Java “light” binding.
 - Slightly less functionality but significantly faster than the initial binding for applications where performance and scalability are more critical.
- Cache 5.0.13 ships with our first JDBC 3.0 driver

ISC and Java Con't



- Adopting Java based philosophy
- Jalapeno
 - Java Persistence with No Mapping
 - Implement Java with no COS
- Java Gateway and Ensemble
 - Leverages the same functionality we needed for Jalapeno
 - Cache to Java for the first time utilizing a JVM accessible in a Cache Process.
- Java IDE Plug-ins (Developer tooling in NetBeans, Eclipse, and IntelliJ)

A look at the Java Industry



Over the last few years Java as a technology has:

- Matured to add features like Generics and Annotations which improve developer productivity.
- Embraced multiple languages like Python, Perl, JavaScript, and Ruby.
- Evolved the enterprise frameworks and specifications to be lightweight and composite to allow for faster development and deployment.
- Focused on building a Java Persistence Architecture specification targeting better Relational Database and Java Integration.
- Pushed the envelope on Development Environments giving developers greater day to day capabilities.

Java Implementations



- We have had success with Java in our customer base thru all the iterations in our technology. Today there are deployed sites using every methodology we have outlined.
 - JDBC
 - NetSmart 1200+ Users at one site
 - Java Gateway
 - MIND Israeli Telecom
 - JMS Message Queuing
 - Light Binding
 - Jalapeno
 - Java Projection
 - Yellow Book 1000+ Users

Java @ InterSystems



Tomorrow we are evaluating how to:

- Further leverage our Jalapeno technology as an answer to the object-relational mapping impedance.
- Implement the JPA standard APIs.
- Pursue other Initiatives as Java Technology continues to evolve.
- Adding Java Technology Product Management - Iran Hutchinson

ISC and .Net



- Cache Factory
- COM Based Legacy from VB
- Web Services
- ADO.Net and ODBC
- Native Provider
- .Net Gateway for Ensemble 2008.1

A look at the .NET Industry



.NET over the last few years has grown in usage on Microsoft platforms. Giving VB, ASP, and C++ technology users options for evolving their technology. We have seen:

- Greater interoperability with Java via new Web Service specifications
- Advances in Visual Studio to aid developer productivity
- Persistence language advances including LINQ
- Broader language support for things like Ruby and Python



- Customers are taking 2 main paths with .Net as the Client
 - SOAP
 - Lothian NHS
 - Nictiz
 - Native Provider
 - Hochtief
- Most implementations seem to be done in C# today with C++ and ASP as the second options.
- .Net Gateway as Client to .Net
 - Leonardo / UK NHS

.NET @ InterSystems



Tomorrow in .NET we are evaluating and works towards:

- Supporting our customers in features needed for their current .NET project work
- Evolving our Managed Provider to support .NET 3.5.x
- Enhance our Web Services for usage from .Net including Security
- As Microsoft develops .Net we will continue to add and modify our tools

Summary



.NET and Java are alive and well at InterSystems. We have tightened our relationship with Microsoft and are involved in the Java community to stay abreast of changes that could impact how and what our customers deploy.

Iran Hutchinson as Product Manager for Java and .Net Technology. He can be reached at iran.hutchinson@intersystems.com

We see both technologies as very important to us and our partners are and committed to helping you succeed in your efforts to leverage these technologies in your solutions

Questions?



Bill McCormick

bill.mccormick@intersystems.com

INTERSYSTEMS