JSR 343 (JMS 2.0) Review for JCP EC

Nigel Deakin

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

ORACLE

Nigel Deakin

- Specification / maintenance lead for JSR 343

- Developer at Oracle
    - GlassFish Message Queue
    - WebLogic JMS

ORACLE

Agenda
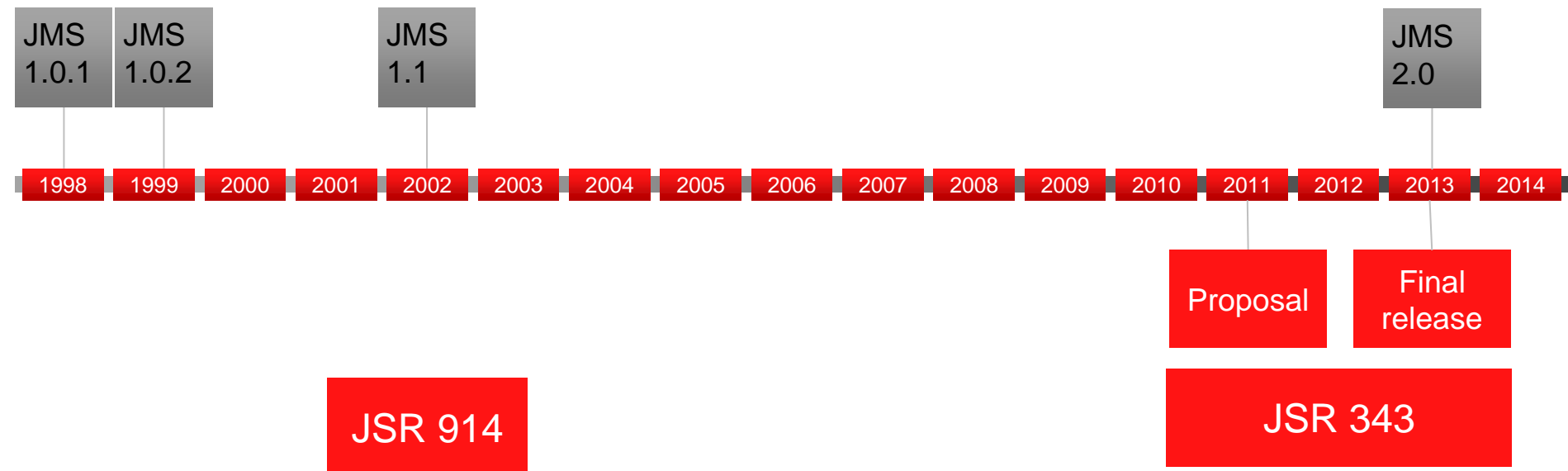
- Introducing JMS and JSR 343

- Community and transparency review

- How things went

- Issues

- Questions

ORACLE

Introducing JMS and JSR 343

ORACLE

Introducing JMS

- Java Message Service
  - An API for accessing enterprise messaging systems from Java programs
- A key component of Java EE
  - A mandatory component of any Java EE full profile application server
- May also be implemented standalone
  - for use by Java SE applications

ORACLE

Introducing JSR 343 (JMS 2.0)

JMS 1.0.1 — 1998
JMS 1.0.2 — 1999
JMS 1.1 — 2002
JMS 2.0 — 2013

| 1998 | 1999 | 2000 | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 | 2007 | 2008 | 2009 | 2010 | 2011 | 2012 | 2013 | 2014 |

Proposal — 2011
Final release — 2013

JSR 914

JSR 343

ORACLE

JSR 343 technical content

- New simplified API (JMSContext)
- Other ease of use features
- Recognition that Java EE, CDI, JCA exist
  - Previous specs completely ignored how JMS should behave in Java EE
  - Injection of JMSContext objects into Java EE applications
  - Specification of resource adapter (but non-mandatory)
- Some new messaging features
  - Delivery delay, async send
- Clarifications and corrections

ORACLE

JSR 343
Community and transparency review

ORACLE

Community and transparency

- JSR 343 used JCP 2.9 but operated transparently from the start

ORACLE

Mailing lists, issue tracker and wiki

- Expert group mailing list (jsr343-experts@jms-spec.java.net)
  - All emails forwarded to community mailing list
- Community mailing list (users@jms-spec.java.net)
  - Anyone allowed to view, subscribe and post, all EG members subscribe
- Issue tracker (java.net/jira/browse/JMS_SPEC)
  - Anyone allowed to submit issues or comments
- Wiki (jms-spec.java.net)
  - Public start point for all JMS 2.0 activities
  - Also used to support expert group discussions
  - Only EG members allowed to update

ORACLE

Source code repository

- Public source code repository (svn)
  - Anyone allowed to read
  - Used for draft API interfaces
  - Used for draft spec document (pdf and docx)
- Public javadocs
  - Used for draft API javadocs

ORACLE

JCP mailing lists

- Did not use JCP observer mailing list
  - Published statement on wiki telling people no need to join it
  - Received periodic requests to join it nevertheless
- Did use JCP expert group mailing list
  - But only very, very occasionally

Reference implementation

Open Message Queue (mq.java.net)

- Standalone JMS provider (passes JMS TCK) and part of GlassFish application server (passes JMS CTS)

- Developed by Oracle

- What is public?

  - Source of promoted builds, build scripts

  - Issue tracker

  - Development one-pagers and specs

  - Community and developer mailing lists

- What is not public?

  - Tests, source of non-promoted builds, internal mailing list

ORACLE

Compatibility tests

- JMS CTS tests JMS in a Java EE application server

- JMS TCK tests JMS standalone

- Developed by Oracle

- Available only to licensees and to EG as required by JCP

JSR 343
How things went

ORACLE

How the spec was developed

- Spec developed almost exclusively using JIRA and email
  - So almost everything was public
- Occasional conference calls to discuss complex details
  - Only expert group invited
- No face-to-face meetings
  - Except at JavaOne
- One-on-one phone calls between all new EG members and spec lead

ORACLE

Expert group membership

Organisations	
Nigel Deakin (lead), Tom Barnes	Oracle
Reza Rahman (resigned)	Caucho
Sastry Maladi	eBay
Matthew White	IBM
Pravesh Pandya	Caucho
Bruce Snyder	Pramati Technologies
Rob Davies Clebert Suconic	RedHat
John Archbold Shivajee Samdarshi	TIBCO

Individuals
John Ament
Adam Bien
Rüdinger zu Dohna
Julien Dubois
John Harby
Andrew Kennedy
Bruce Snyder
Nick Wright

ORACLE

Expert group participation

- 1066 emails sent to jsr343-experts@jms-spec.java.net
- 50% of emails sent by spec lead
  - 541
- 40% of emails sent by the 4 most active members
  - 127, 110, 105, 84
- 10% of emails sent by the remaining 15 members
  - 13, 4, 7, 8, 3, 10, 6, 3, 11, 5, 5, 16, 1, 1, 1

ORACLE

JMS spec community participation

- 97 community members subscribed to users@jms-spec.java.net)
- 10 community members contributed to users@jms-spec.java.net
  - 37, 22, 17, 4, 3, 2, 2, 1, 1,1 (90 in total)
- Several community members logged new JIRA issues
- Several community members commented on JIRA issues

ORACLE

Conference sessions

All well-attended

- JavaOne 2011 BOF (with Reza Rahman and Clebert Suconic)
- Devoxx 2011 Session
- JavaOne 2012 Session
- JavaOne 2012 BOF (with Nick Wright)
- JavaOne 2013 Session
- JavaOne 2013 BOF (with John Ament)

ORACLE

Issues

ORACLE

Not enough expert group participation

- Even from major vendors
- Several expert group members played no part after joining

ORACLE

Is a separate expert group redundant?

- Any community member can read and take part in all discussions
- Little gained by joining the EG
  - except perhaps kudos for book authors, consultants etc
- Is expert group an unnecessary barrier to participation?
  - At least when EG members are not contributing code
- EG membership useful public gesture of support from vendors

More scope to use community expertise

- Reluctance of community to bring forward their own detailed proposals, especially in areas requiring specialist knowledge
- Example
  - Community: "The spec should do something about XXXX"
  - Spec lead: "What do you think the spec should say?"
  - Community: <silence>
- Where "XXXX" is web APIs, AMQP, etc
- Where community members did come forward with proposals and expertise, they were very influential (e.g. CDI integration)

ORACLE

Participation tools could be improved

- jcp.org facilities (e.g. mailing lists) inadequate and not used
  - Existence still confused people
- java.net mailing lists generally adequate, but
  - Inadequate web interface. Expecting people to subscribe to a mailing list to keep informed is out of date.
  - Using two separate email aliases is complicated to manage for both EG and community members

ORACLE

Backward compatibility

- Difficult to resolve ambiguities in spec
  - without breaking backward compatibility
- Difficult to define behaviour which previous spec left undefined
  - might require implementations to introduce incompatible changes

What would I do differently next time?

- Hold regular conference calls to stimulate participation
- Reach out directly to inactive vendors on the expert group

Questions?

ORACLE