



# Qisda JSRs: Current Status and Further Action

Presented by Kay Glahn

26 September 2008

## Overview

- Java ME evolution process is stalled due to Qisda JSRs (e.g. completeness of MSA and API set)
- Java ME is not able to keep up with competitive technologies because important building blocks for a platform are missing
- Investment of the whole industry is under risk (to some extent)
- Companies do not file new JSRs due to the unresolved issue with Qisda JSRs
- Urgent action is necessary to make progress and prevent similar situations in the future



## Current Status

- Qisda has **relinquished** spec leadership for all JSRs. What does this mean from a legal perspective:
  - \_ Does Qisda still own the *IPR of the specifications*??
  - \_ Can the JSRs been taken over by a new spec lead? What is the meaning?
    - Could a new spec lead
      - = build on a half-finished specification? Progress it and produce TCK&RI?
      - = Take over a completed specification and produce TCK&RI?
  - \_ Is any company allowed to file a new JSR with the same or overlapping focus?
  - \_ Is any ballot necessary by the EC in order to decide if the JSR will still be alive?
- **TCK & RI IPR** is still owned by Qisda and the insolvency administrator of BenQ Mobile Germany:
  - \_ Do they still own the *IPR of the TCK&RI* after relinquishing spec leadership?
  - \_ Is any of the two parties willing and allowed to sell the IPR?
  - \_ Is any other model feasible (e.g. open source)?



# Affected JSRs

- JSR 229 Payment API
  - \_ Stage: Final (Maintenance Release)
  - \_ Comment: Was part of MSA 1.0 and had to be removed from MSA 1.1
- JSR 230 Data Sync API
  - \_ Stage: Proposed Final Draft
- JSR 246 Device Management API
  - \_ Stage: Early Draft Review
- JSR 253 Mobile Telephony API
  - \_ Stage: Final
  - \_ Comment: Motorola is spec lead but doesn't own the IPR
- JSR 259 Ad Hoc Networking API
  - \_ Stage: Early Draft Review
- JSR 266 Unified Message Box Access API
  - \_ Stage: Early Draft Review



# Solution Options

1. File new JSR with same or overlapping focus
  - a. Recycle parts of the existing JSR if the legal situation allows.
  - b. Bring together the former expert group and reuse at least their know how.
  - c. Start the JSR from scratch (no prior knowledge assumed).
  
2. Take over spec leadership of an existing JSR
  - a. Use the existing specification as a basis if the legal situation allows.
  - b. Develop the TCK&RI from scratch.
  - c. Buy the IPR of the existing TCK&RI from Qisda.
  - d. Encourage Qisda to contribute the TCK&RI as open source which would become worthless anyway if any of the other options is used.



## Further Actions

- JCP Process needs a redesign to prevent such or similar situations in the future.
  - Java ecosystem can only be competitive if new innovation can be transferred smoothly into specifications without any legal hassles.
  - A fall-back mechanism has to be established for the case that TCK&RI implementers are not able or willing to support their software any more.
  - Even if these kinds of situations don't occur frequently it can happen again at any time. (low probability doesn't exclude that it can happen 7 times next week)
- The JSPA could provide a legal framework to handle such a situation.
  - Requiring to provide the TCK&RI always as open source under an appropriate license would solve the issue.
  - A mechanism could be established which still allows TCK&RI implementers to get a return on their investment even if the source code is available as open source.



# What Happens If We Don't Resolve The Situation

- Lack of certain APIs or hindrance to specify new APIs of same or similar scope will entail the following:
  - \_ The major effort of MSA cannot be rounded off by adding relevant APIs
  - \_ Companies may create own, proprietary APIs in Java.
    - = More fragmentation, more segmented developer communities,
    - = Smaller communities per solution,
    - = Less market reach for the applications as available only on certain mobile devices.
    - = Where carriers want to influence the specification of the API, this results into a bigger burden for the mobile industry: effort on the side of the carriers, effort/changes on the side of the vendors. → Makes Java as the technology of choice less attractive.
  - \_ Companies exploring alternative technologies, e.g. web runtime APIs.
    - = This is happening.
    - = It's taking away funds otherwise allocated to progressing the Java ecosystem and allocating them to new players in a different ecosystem.



Leaving the situation as is: not a good option. "Java ecosystem suicide"



## ... Therefore

- Is the JCP capable to resolve this?
- In the mobile business, nowadays **days** count.
- ... and windows of opportunity close (per definition)







**Kay Glahn**

**Consultant Mobile Service Architecture**

Vodafone Group R&D

Vodafone Group Services GmbH

Chiemgaustr. 116

81549 Munich

Germany

T + 49 89 95410 0

F + 49 89 95410 111

[kay.glahn@vodafone.com](mailto:kay.glahn@vodafone.com)

[www.vodafone.com](http://www.vodafone.com)

