



Best Design Practices Using the Qt Runtime SDK (using QML)

Michael Tims – Esri

Session Agenda

- **Bundled SDK Components**
 - Toolkit
 - Extras
- **Modularize your QML Components**
- **JavaScript code in QML**
 - Object ownership and garbage collection
- **Property binding**
 - Property notifications
 - Model/View
- **Testing and Deployment**
- **Q&A**

Bundled SDK Components

Use what's already available to you

- **Toolkit - Controls and Dialogs**
 - Reuse or subclass existing components
- **ArcGIS Extras plugin for Qt**
 - High-demand APIs exposed to QML
 - File I/O
 - Application settings
 - UI scale factor

Modularize your QML Components

Break apart UI from business logic

- **Keep your UI in separate QML files**
 - Declare component as a local type
- **Access components through custom properties**

JavaScript code in QML

QML is a JavaScript-based language

- **Write or use existing JavaScript code in QML**
 - Write custom JavaScript functions
 - Use common JavaScript objects (Math, Date, etc...)
- **Be aware of object ownership**
 - QML-declared components are owned by the QML Engine/JS Engine
 - Objects returned from `ArcGISRuntime.createObject()` are given JS ownership

Property binding

The basis for property notifications

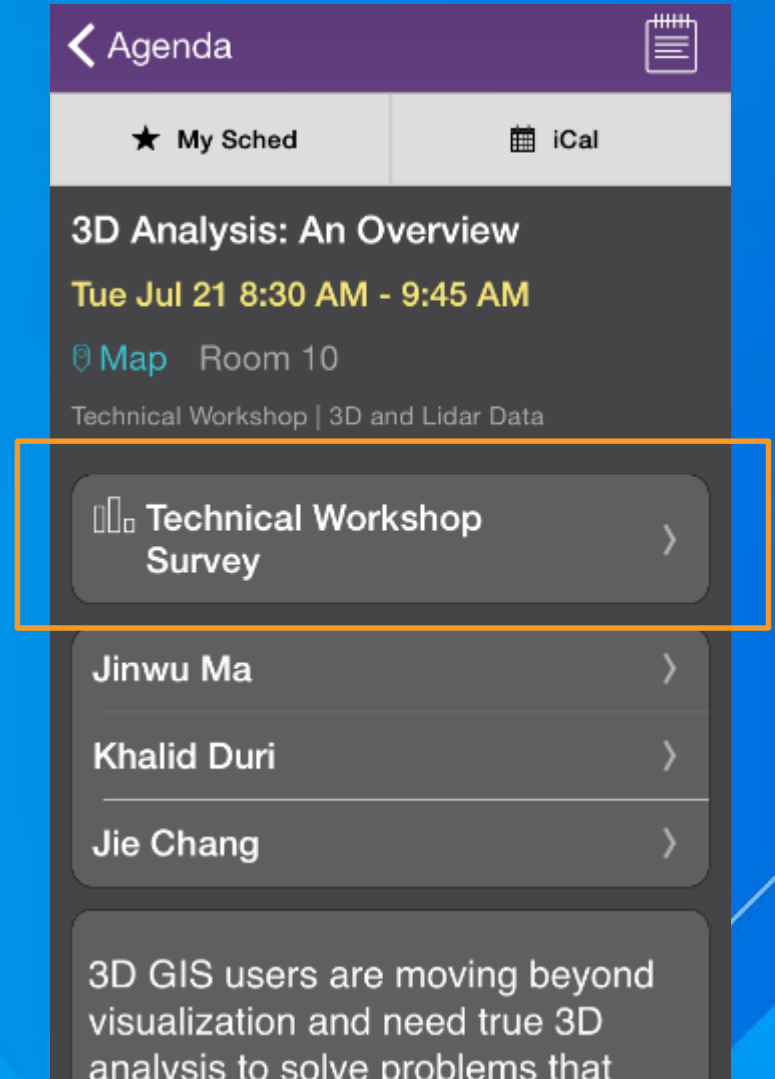
- **UI updates when property changes**
 - Property notifies any other bound property that it has changed
- **Create property bindings from JavaScript**
 - `property = Qt.binding(function() { return calculated_value })`
- **Models and Views**
 - Models notify the view to update when data changes
 - Binding list properties to QML views

Testing and Deployment

- Test and debug your app directly from Qt Creator
- Deploy your app with local data

Thank you...

- Please fill out the session survey in your mobile app
- Select “Best Design Practices Using the Qt Runtime SDK (using QML)” in the Mobile App
 - Use the Search Feature to quickly find this title
- Click “Demo Theater Survey”
- Answer a few short questions and enter any comments



Questions?



Understanding our world.