Structure Concepts, Developer's Perspective

This article provides an introduction to the main concepts used in Structure. Before starting your work on integration with Structure, please familiarize yourself with these concepts.

1. Basic Concepts Overview

Concept	Short Definition	API Classes to Check
Structure	A named container for a hierarchical list.	`Structure`, `StructureManager`
Forest	A hierarchical list.	`Forest`, `ForestService`
Row	A row is a unique, atomic element of a forest.	`StructureRow`, `RowManager`
Item	An item is a user-level object (like Issue) that is displayed in a row.	`ItemIdentity`, `CoreIdentities`
Attribute	An attribute provides values of a certain type and meaning for forest rows.	`AttributeSpec`, `StructureAttributeService`
Column	A column loads one or more attributes and displays information about forest rows.	`ViewSpecification`
View	A view is a named collection of columns.	`StructureView`, `StructureViewManager`

Important points:

- **Structures** are the main entities provided by Structure add-on. A structure has name and other attributes, like description, and it also has content, represented by a **forest*.***
- A forest represents a structure's content. But it can also represent a result of a query or a hierarchical list received or stored somewhere else.
- Forest contains **rows**. Forest content is actually a list of pairs (`row ID, depth`).
- A row has a numeric ID that uniquely identifies it in a forest. A forest may not contain the same row twice. (Although a row may be present in different forests.)
- When users look at a structure, they see a grid – each row in that grid is represented by a Structure's row.
- A row refers to an **item**. An item is an abstraction for everything that can be placed into a forest – issues, folders, projects, users are all items, from Structure's perspective.
- An item has **item identity** – something that uniquely identifies that item on a JIRA instance.
- An item also has **attributes** – some values with associated meaning, which Structure and its extensions can provide and that can be shown to the user.

2. A Note on Extensibility

Structure is built with extensibility in mind. It is possible for a separate add-on to add new item types, attributes, columns and other extensible elements to Structure, at runtime.