Open Source
Tooling Group

Capability Map

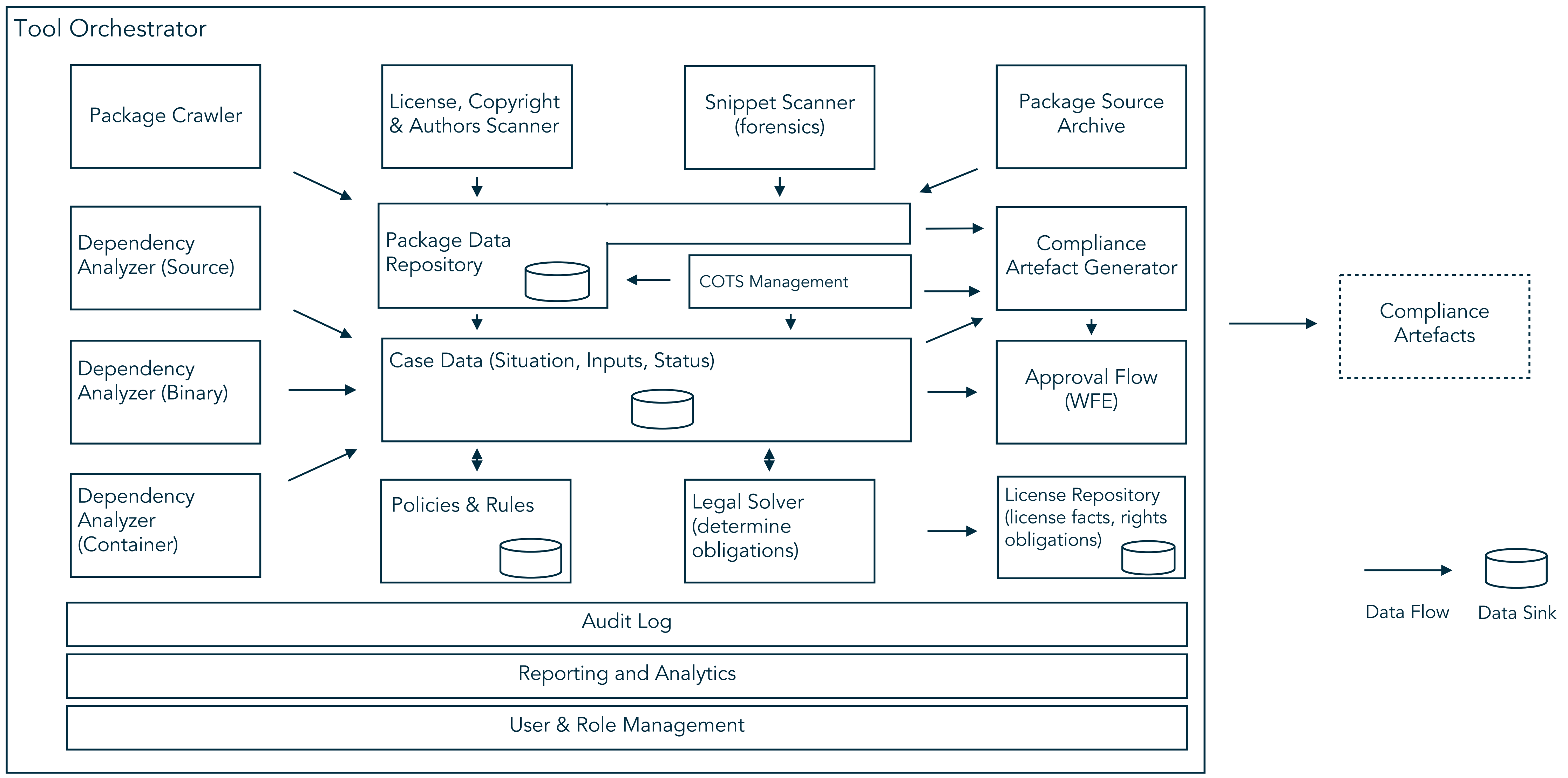
OC Tooling Reference Workgroup - v1.3.2

v1.3.2 by Dr. Peter Ellsiepen (ESA) & Jan Thielscher (TrustSource), 11.1.2021

Changelog

Version	Date	by	Comments/Changes
1.2	3.12.19	Jan, Peter	Initial draft
1.3	6.12.19	Jan	Rename Case Data => Situation Data, delete „Compliance Artefacts“ as capability, change Mission of Snippet scanner
1.3.1	11.1.21	Jan	Review spelling, add some Readme's in the surrounding, review & harmonize definitions
1.3.2.	11.1.21	Jan	Added a few samples for capability mapping

ToolChain Capabilities - Overview


Tool Orchestrator
Package Crawler
License, Copyright & Authors Scanner
Snippet Scanner (forensics)
Package Source Archive
Dependency Analyzer (Source)
Package Data Repository
COTS Management
Compliance Artefact Generator
Compliance Artefacts
Dependency Analyzer (Binary)
Case Data (Situation, Inputs, Status)
Approval Flow (WFE)
Dependency Analyzer (Container)
Policies & Rules
Legal Solver (determine obligations)
License Repository (license facts, rights obligations)
Data Flow
Data Sink
Audit Log
Reporting and Analytics
User & Role Management

ToolChain Capabilities - Package Crawler/Finder

Mission	• Research information on (new) components such as locate the repository, current and former versions, project homepage and viability information
Responsibilities	• Collect and provide accurate information about the component • Alert, if component can't be matched/found
Tasks	• Scan package managers for new packages or versions of packages • Collect package data • Transfer data into package repository
Input	• Component descriptor or component name
Output	• Component Information, such as: source repository url, version history, branches, commit count, stars, last commit date, etc.
Comments	

ToolChain Capabilities - Dependency Analyzer (Source)

Open Source
Tooling Group

Mission	• Provide composition analysis of software to be built from these sources
Responsibilities	• Determine all packages and dependencies used to build the software • Allow to stop a CI/CD chain, if violations occur
Tasks	• Integrate with build process (CI/CD) • Determine composition (complete Bill of Materials) • Provide output for further analysis, e.g. as SPDX • Provide link between scanned source and BoM information, e.g. Commit ID
Input	• Build description, e.g. POM or requirements.txt
Output	• Bill of Materials (BoM) for particular build
Comments	Analysis and dependency resolution is highly language specific. Thus a language specific implementation might be required

ToolChain Capabilities - Dependency Analyzer (Binary)

Mission	• Provide composition analysis of a software binary
Responsibilities	• Determine all packages and dependencies used within this binary • Allow to stop a CI/CD chain, if violations occur
Tasks	• Download binary (if required) • Unpack binary • Assess content and determine used packages/components • Collect information and assemble Bill of Materials • Provide Bill of Materials (e.g. as SPDX) • Provide link between BoM and scanned artefact, e.g. binary repo ID
Input	• Binary or link to binary location
Output	• Bill of Materials (BoM) for particular binary
Comments	• Hash to identify the binary scanned should be generated and archived

ToolChain Capabilities - Depdendency Analyzer (Container)

Open Source
Tooling Group

Mission	• Provide composition analysis of a container
Responsibilities	• Determine all packages and dependencies used within this container • Allow to stop a CI/CD chain, if violations occur
Tasks	• Download container (if necessary) • Assess container content/structure and determine used packages/components • Collect information and assemble Bill of Materials • Provide Bill of Materials (e.g. as SPDX) • Provide link between BoM and scanned container, e.g. Repo + image ID + tag
Input	• Container or link to container location
Output	• Bill of Materials (BoM) for particular container
Comments	▪ Hash to identify the scanned container should be generated and archived

ToolChain Capabilities - License, Copyright & Authors Scanner

Open Source Tooling Group

Mission	• Precise scanning of sources to determine exact situation for compliance proper declarations
Responsibility	• Ensure correctness of compliance information
Tasks	• Identify copyright statements • Identify authors • Identify effective licenses
Input	• Repository or file(s) to scan
Output	• List of effective and declared licenses with links into code • List of copyright statements with links into code • List of author information with links into code
Comments	• TODO: Clarify granularity required to differentiate between author, commiter and copyright holder

ToolChain Capabilities - Snippet Scanner

Open Source
Tooling Group

Mission	• Identify origin of sources
Responsibility	• Ensure source code is free from copyright infringements due to copying routines or third party code
Tasks	• Scan sources for known snippets • Provide scan results
Input	• Repository or file(s) to scan
Output	• List of potential infringements with links to potential matches (e.g. in existing OSS) • Weighting/ordering of potential matches
Comments	• TODO: Discuss whether snippet similarity and file similarity should be treated the same. While snippet similarity often is seen critical due to many false positives, file similarity can be pretty relevant, especially for languages such as C/C++ or Python

ToolChain Capabilities - Package Data Repository

Open Source
Tooling Group

Mission	• Collect package information and clearing data on packages
Responsibility	• Single point of truth for package information
Tasks	• Store packages data • Support composition analysis (verification of dependency analysis) • Provide search capabilities to identify existing packages • Support authentication/authorization to ensure responsible data handling/editing
Input	• Package data and metadata (if known)
Output	• Package data and metadata, including package type (e.g. OSS, COTS, internal) and completion/verification status of associated metadata • Containment structures (consists of) • Dependency structures (depends on) • Optional: relate known vulnerability information (not OSC specific, but a good place)
Comments	• TODO: Clarify role or repo in relation to the archive function. SW360 comes as archive, which actually could also be served by Git or binary repositories. Thus adding an archive function here might just duplicate the code • TODO: Clarify unique identification of OSS pkgs, e.g. package URL, to be mandatory

ToolChain Capabilities - Situation Data (Structure of Solution, Circumstances, etc.)

Open Source
Tooling Group

Mission	• Provide bracket for all compliance relevant information that is not directly related to source of a product / distribution item
Responsibility	• Ensure completeness of product documentation
Tasks	• Collect all product specific information, including package change & linkage status (via history) • Follow the release cycle of a particular product, e.g. approvals • Organize access rights and assign roles • Build canvas for reporting and analysis of a given composition & in a given situation
Input	• Bill of Materials (BoM) • External components, e.g. runtime environments, middleware or resources • Participants / Stakeholders
Output	• Status Overview • History of events • Reporting
Comments	

ToolChain Capabilities - Policies & Rules

Open Source
Tooling Group

Mission	• Document context and evolution of the context of a project
Responsibility	• Track all relevant changes in the project environment
Tasks	• Document legal circumstances, e.g. commercial aspects, trade secrets, export aspects or IP protection requirements, etc. • Document changes in project specific black lists or whitelists • Track changes • Allow managing groups of projects with consistent policies & rules
Input	• Legal requirements • Black- and whitelists • Project specific roles or policies
Output	• History of changes
Comments	• TODO: how to capture policies & rules in a form that allows automation/repetition

ToolChain Capabilities - COTS Management

Mission	• Manage Commercial-Off-The-Shelf (COTS) and infrastructure packages of a solution
Responsibility	• Allow tracking of composition as well as 3rd party vulnerability and compliance tracking • Collect and provide data for 3rd party or infrastructure packages
Tasks	• Provide place to store 3rd party package and license information • Allow to assemble reports like SOUP-lists • (Review 3rd party assemblies for known vulnerabilities)
Input	• Package data and metadata (if known) • Binary scan information (BoM)
Output	• Package data and metadata (updated)
Comments	• TODO: Build consensus on whether to include the vulnerability information or not. It is not required for compliance purposes

ToolChain Capabilities - Legal Solver

Open Source
Tooling Group

Mission	• Determine legal rights and obligations resulting from the usage of the listed packages within the project context
Responsibility	• Provide compliance requirements
Tasks	• Assess license information from all packages (recent BoMs, infrastructure and COTS) • Determine license obligations • Identify effective licenses
Input	• Composition analysis of all project related packages, their status and licenses • Legal circumstances and requirements
Output	• List of legal obligations by package and mitigation hints
Comments	• Independent from package status the analysis results may vary depending on changes in the circumstances. Thus analysis results should be versioned to allow allocation to related circumstances.

ToolChain Capabilities - License Repository

Open Source
Tooling Group

Mission	• Capture and archive legal information about licenses
Responsibility	• Manage and provide legal information about licenses
Tasks	• Capture all license information including derived requirements • Provide environment to allow license analysis • Track license data changes • Provide reference for original license texts
Input	• License data
Output	• License data (updated)
Comments	• Could be combined with legal solver, but we decided to provide as separate capability. A solver requires the repository, but the solver also could be a human worker.

ToolChain Capabilities - Compliance Artefact Generator

Open Source
Tooling Group

Mission	• Support provisioning of compliance documentation
Responsibility	• Ensure legally compliant documentation
Tasks	• Generate documentation according to requirements • Support Compliance Managers in completing tasks • Provide documentation parts, e.g. written offer, license texts, copyrights, modification statement, etc. • Link documentation with documentation objects (version management)
Input	• List of versioned packages to be documented (BoMs) • Legal requirements with respect to particular circumstances
Output	• Stub with all documentation requirements • Pre-assembled stub with all existing information (e.g. from repository) • Identified TODOs for missing bits
Comments	• TODO: We might consider to define a specific output format (e.g. PDF, JSON, SPDX, etc.)

ToolChain Capabilities - Approval Flow

Open Source
Tooling Group

Mission	• Help decentralising compliance work through approval
Responsibility	• Provide approval flow appropriate for audit
Tasks	• Track all legally relevant changes to products and packages • Identify authors of change • Provide compliance status and overview • Allow to approve or reject an approval request • Document/archive all decisions (auditing)
Input	• Approval request for (list of packages, legal situation, compliance documentation)
Output	• State of compliance analysis for approval request • Approval / Rejection documentation
Comments	• The approval by a dedicated, skilled resource (Compliance Manager) combined with the automation support for all prior steps reduces the need for Compliance Managers

ToolChain Capabilities - User & Role Management

Mission	• Provide role based authorization
Responsibility	• Authenticate users • Manage roles and authorizations • Assign users to roles
Tasks	• Identify users (Login, oAuth, MFA) • Manage roles and related authorizations (permissions assigned to roles) • Manage API Keys
Input	• Users • Roles • Assignments (user to project, etc.)
Output	• Access tokens
Comments	• TODO: Discuss whether this shall be a capability. As information about non-compliance might be critical aspect I would suggest to include it. But from a pure functional point of view, this seems not to be required

ToolChain Capabilities - Audit Log

Mission	• Maintain log of changes and user actions
Responsibility	• Ensure confirmability of configuration changes • Ensure tracing and archiving of all user actions/decisions for auditing purposes
Tasks	• Track user activity and changes in settings, especially legal settings • Track and archive user decisions and related context to enable auditing
Input	• User actions / events
Output	• History of changes with actors • Transparency
Comments	

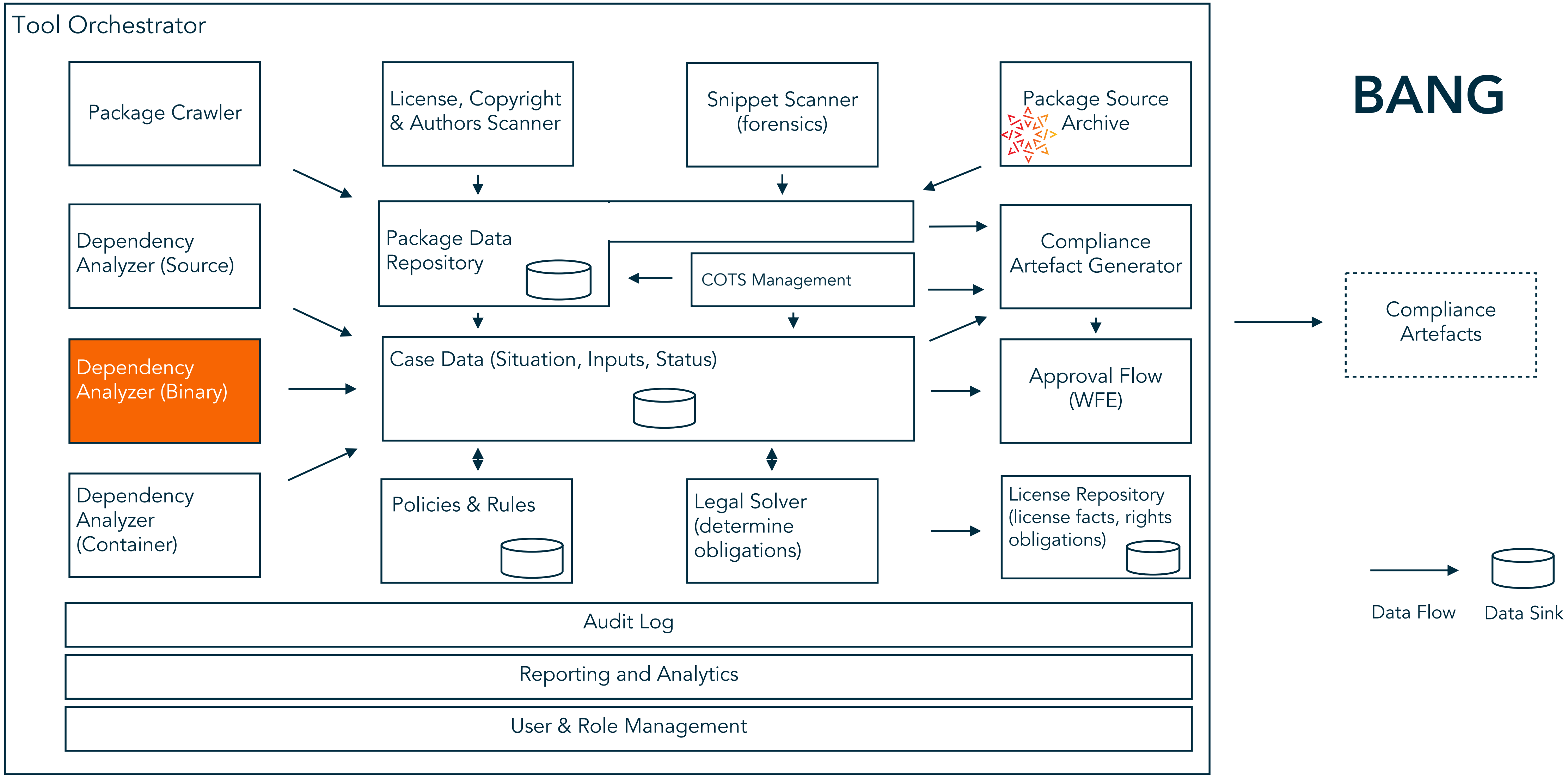
ToolChain Capabilities - Reporting & Analytics

Mission	• Visualize work, efforts and success of compliance initiative
Responsibility	• Measure compliance related activity • Provide insights into state of portfolio
Tasks	• Provide lists and and insights
Input	• Report specific configuration
Output	• Reports • Transparency
Comments	• TODO: Discuss whether we want to define specific reports that shall be supported

ToolChain Capabilities - Tool Orchestrator

Mission	• Realize overall compliance workflow and machinery
Responsibility	• Arrange combination of tools to cope with compliance challenge • Handle handover between capabilities
Tasks	• Trigger events
Input	• Events
Output	• Events
Comments	• TODO: Discuss whether we want to define specific events in an underlying flow

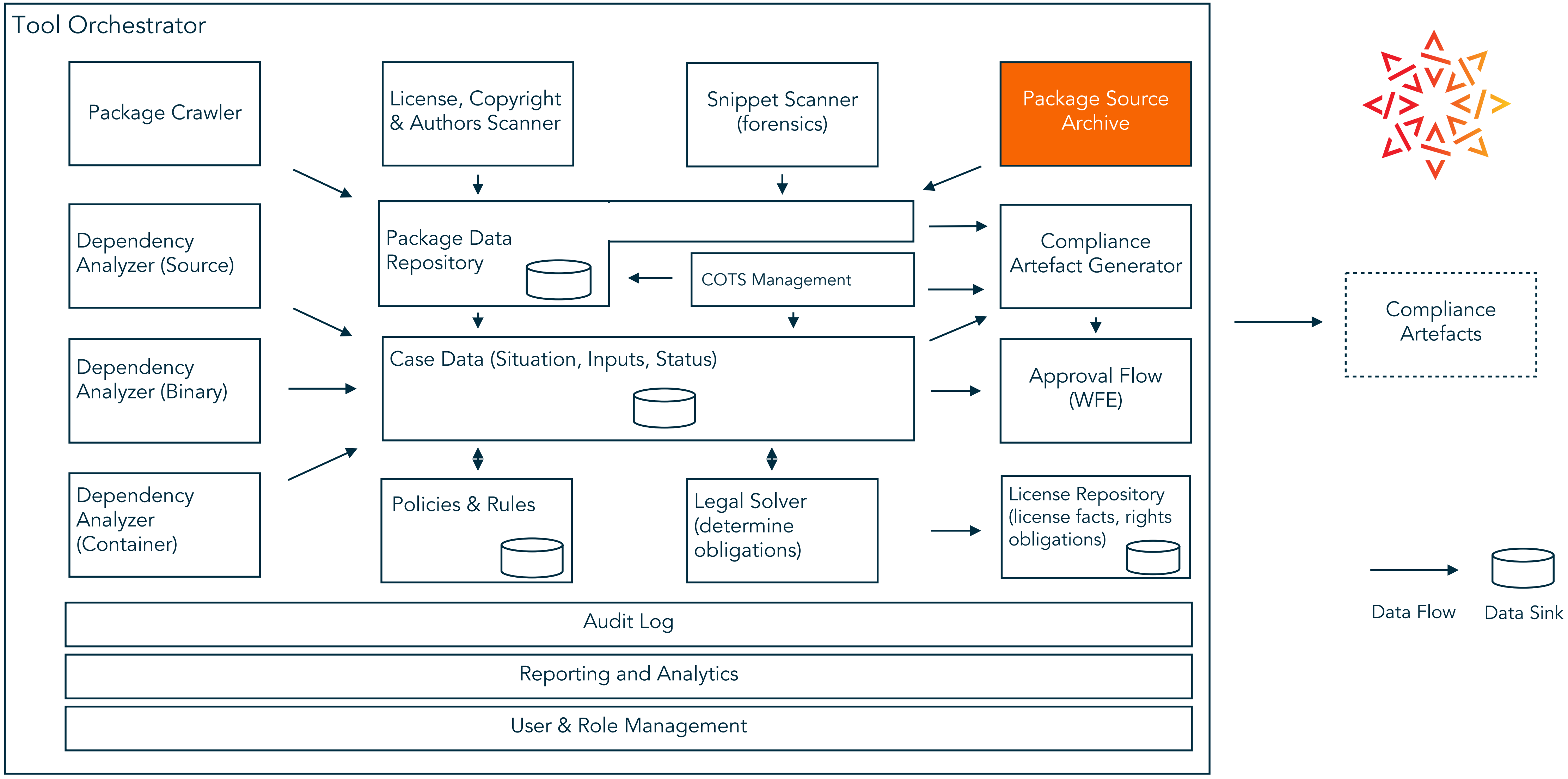
ToolChain Capabilities (v1.3.1) – Mapping of Tools (example BANG)



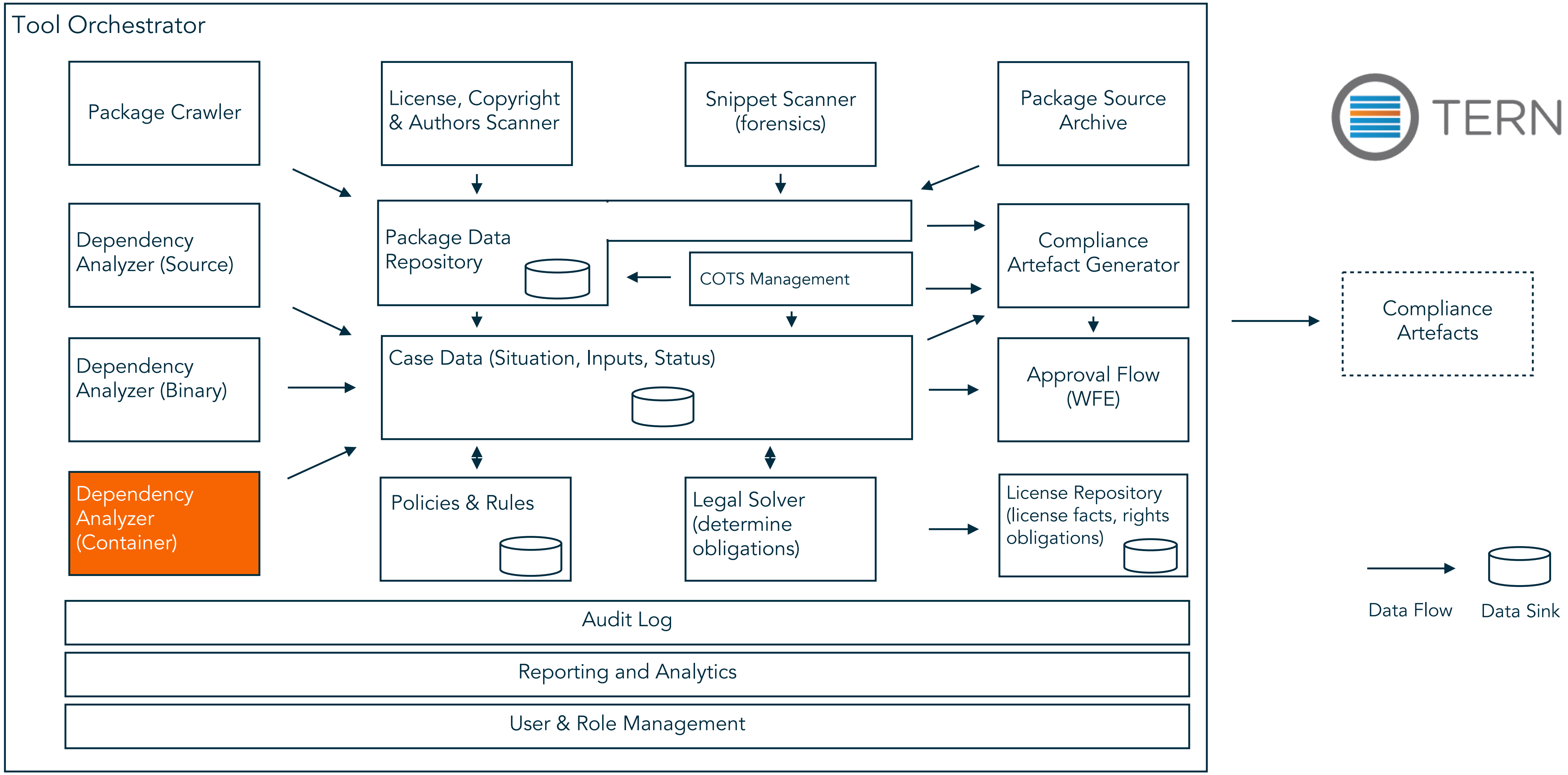
Tool Orchestrator
Package Crawler
License, Copyright & Authors Scanner
Snippet Scanner (forensics)
Package Source Archive
Dependency Analyzer (Source)
Package Data Repository
COTS Management
Compliance Artefact Generator
Dependency Analyzer (Binary)
Case Data (Situation, Inputs, Status)
Approval Flow (WFE)
Compliance Artefacts
Dependency Analyzer (Container)
Policies & Rules
Legal Solver (determine obligations)
License Repository (license facts, rights obligations)
Audit Log
Reporting and Analytics
User & Role Management
Data Flow
Data Sink

BANG

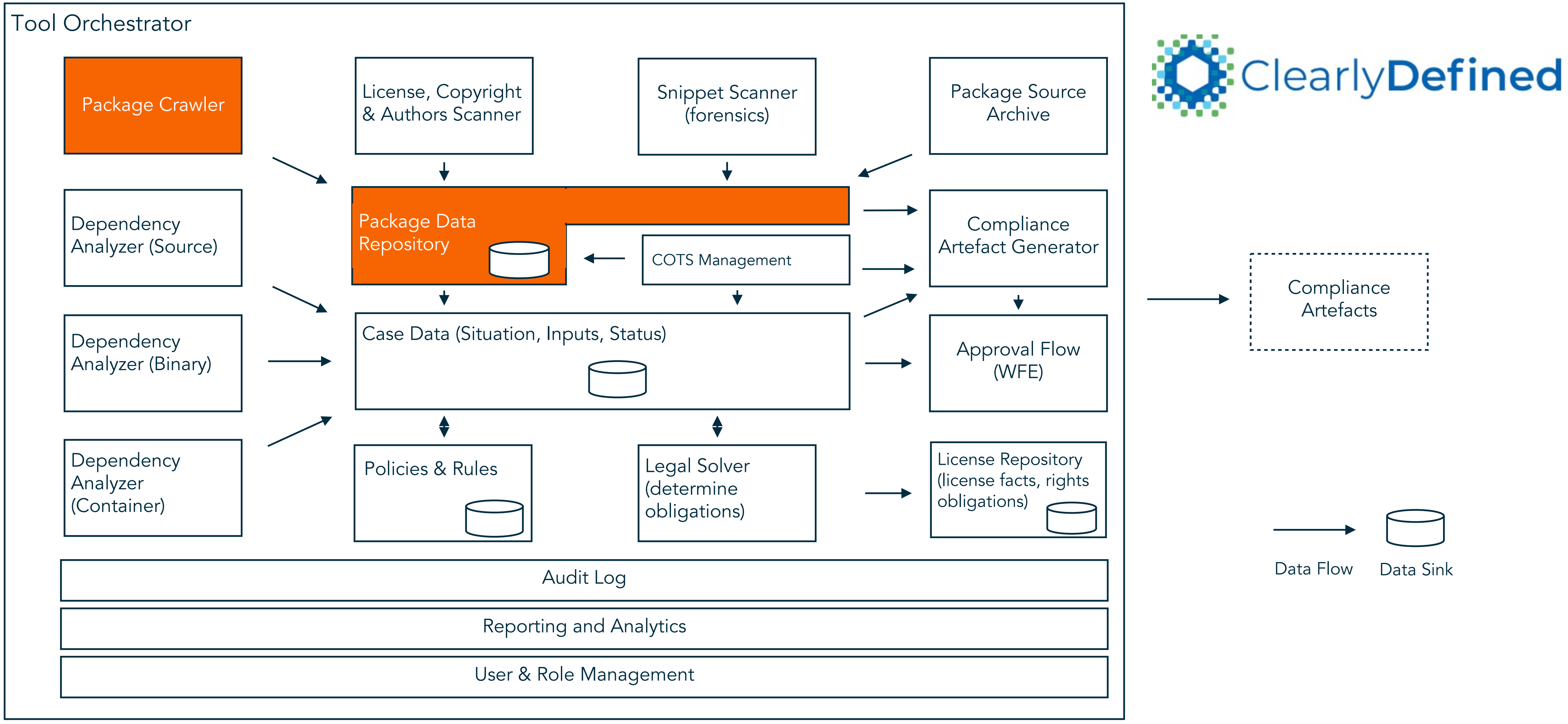
ToolChain Capabilities (v1.3.1) – Mapping of Tools (example Software Heritage)


Tool Orchestrator
Package Crawler
License, Copyright & Authors Scanner
Snippet Scanner (forensics)
Package Source Archive
Dependency Analyzer (Source)
Package Data Repository
COTS Management
Compliance Artefact Generator
Compliance Artefacts
Dependency Analyzer (Binary)
Case Data (Situation, Inputs, Status)
Approval Flow (WFE)
Dependency Analyzer (Container)
Policies & Rules
Legal Solver (determine obligations)
License Repository (license facts, rights obligations)
Audit Log
Reporting and Analytics
User & Role Management
Data Flow
Data Sink

ToolChain Capabilities (v1.3.1) – Mapping of Tools (example TERN)

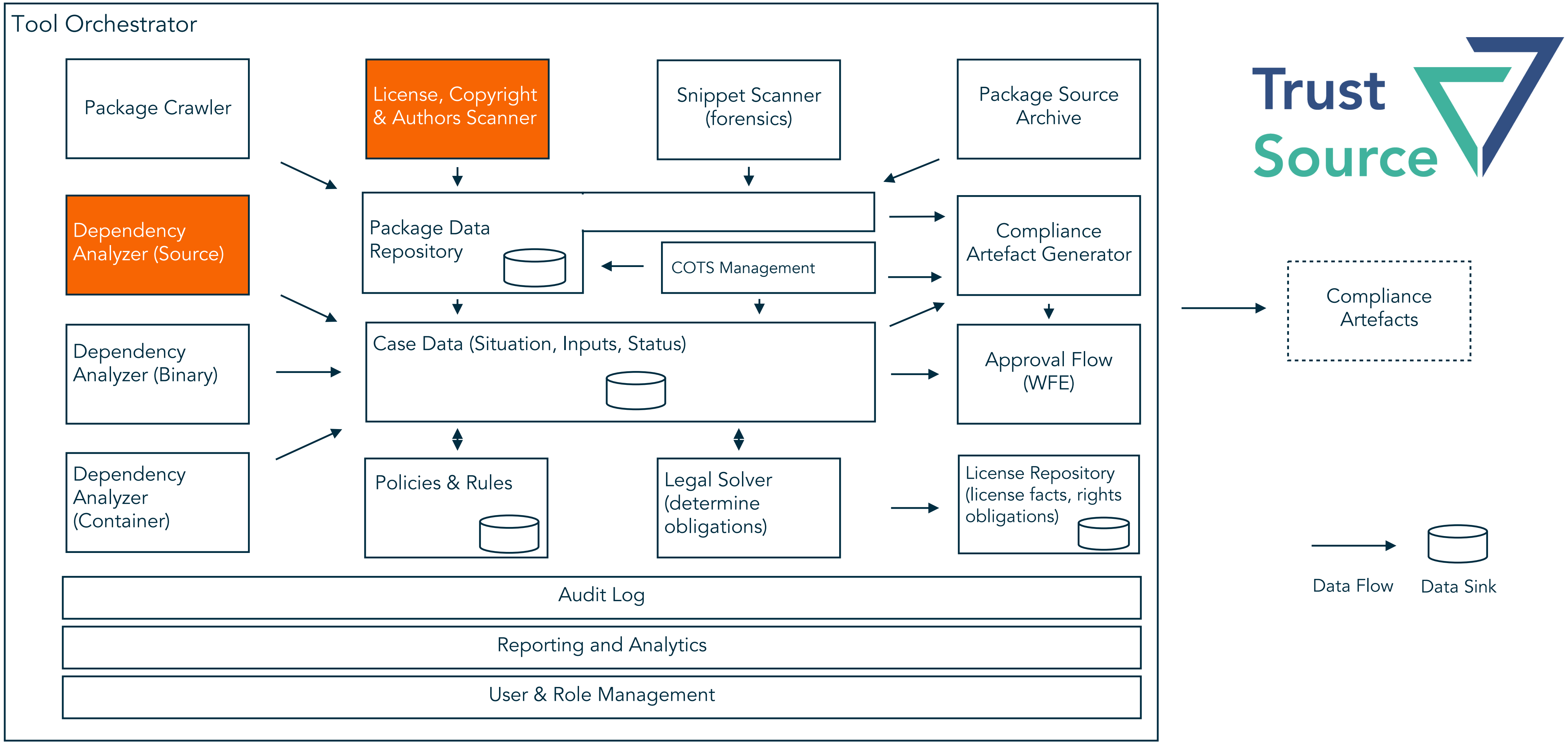

Tool Orchestrator
Package Crawler
License, Copyright & Authors Scanner
Snippet Scanner (forensics)
Package Source Archive
Dependency Analyzer (Source)
Package Data Repository
COTS Management
Compliance Artefact Generator
Dependency Analyzer (Binary)
Case Data (Situation, Inputs, Status)
Approval Flow (WFE)
Compliance Artefacts
Dependency Analyzer (Container)
Policies & Rules
Legal Solver (determine obligations)
License Repository (license facts, rights obligations)
Audit Log
Reporting and Analytics
User & Role Management
TERN
Data Flow
Data Sink

ToolChain Capabilities (v1.3.1) – Mapping of Tools (example ClearlyDefined)



Tool Orchestrator
Package Crawler
License, Copyright & Authors Scanner
Snippet Scanner (forensics)
Package Source Archive
Dependency Analyzer (Source)
Package Data Repository
COTS Management
Compliance Artefact Generator
Compliance Artefacts
Dependency Analyzer (Binary)
Case Data (Situation, Inputs, Status)
Approval Flow (WFE)
Dependency Analyzer (Container)
Policies & Rules
Legal Solver (determine obligations)
License Repository (license facts, rights obligations)
Audit Log
Reporting and Analytics
User & Role Management
ClearlyDefined
Data Flow
Data Sink

ToolChain Capabilities (v1.3.1) – Mapping of Tools (example TrustSource Scanners)



Tool Orchestrator
Package Crawler
License, Copyright & Authors Scanner
Snippet Scanner (forensics)
Package Source Archive
Dependency Analyzer (Source)
Package Data Repository
COTS Management
Compliance Artefact Generator
Compliance Artefacts
Dependency Analyzer (Binary)
Case Data (Situation, Inputs, Status)
Approval Flow (WFE)
Dependency Analyzer (Container)
Policies & Rules
Legal Solver (determine obligations)
License Repository (license facts, rights obligations)
Data Flow
Data Sink
Audit Log
Reporting and Analytics
User & Role Management