


STUNIR Phase 2 SPARK Migration

Build System & Configuration

Generated: January 29, 2026
Phase: 2
Status:  COMPLETE

Overview

Phase 2 migrates the STUNIR build system and configuration utilities from Python/Shell to Ada SPARK. This phase adds 6 critical components:

Component	Purpose	Source
Epoch Manager	Deterministic epoch selection	tools/epoch.py
Toolchain Discovery	Host toolchain scanning	scripts/discover_toolchain.sh
Config Manager	Build configuration	scripts/build.sh
Dependency Resolver	Dependency acceptance	tools/dep_receipt_tool.py
Receipt Manager	Receipt generation	tools/record_receipt.py
Build Orchestrator	Build pipeline coordination	scripts/build.sh

Architecture

```

core/
├── epoch_manager/
│   ├── epoch_types.ads      -- Epoch type definitions
│   ├── epoch_types.adb
│   ├── epoch_selector.ads   -- Epoch selection logic
│   └── epoch_selector.adb
├── toolchain_discovery/
│   ├── toolchain_types.ads  -- Tool entry types
│   ├── toolchain_types.adb
│   ├── toolchain_scanner.ads -- Host scanning
│   └── toolchain_scanner.adb
├── config_manager/
│   ├── build_config.ads     -- Configuration types
│   ├── build_config.adb
│   ├── config_parser.ads    -- Configuration parsing
│   └── config_parser.adb
├── dependency_resolver/
│   ├── dependency_types.ads  -- Dependency types
│   ├── dependency_types.adb
│   ├── dependency_resolver.ads -- Resolution logic
│   └── dependency_resolver.adb
├── receipt_manager/
│   ├── receipt_types.ads    -- Receipt types
│   ├── receipt_types.adb
│   ├── receipt_generator.ads -- Receipt creation
│   └── receipt_generator.adb
└── build_system/
    ├── build_orchestrator.ads -- Pipeline coordination
    └── build_orchestrator.adb

```

Component Details

1. Epoch Manager

Files: epoch_types.ads/adb , epoch_selector.ads/adb

Purpose: Provides deterministic epoch selection with priority-based fallback.

Key Types:

- Epoch_Value - Unix timestamp (0 to $2^{63}-1$)
- Epoch_Source - Source enumeration (env, derived, git, zero)
- Epoch_Selection - Selection result with value, source, and determinism flag

Key Functions:

- Select_Epoch - Main epoch selection with priority
- Parse_Epoch_Value - String to epoch parsing
- Derive_Epoch_From_Digest - Epoch from spec hash

2. Toolchain Discovery

Files: toolchain_types.ads/adb , toolchain_scanner.ads/adb

Purpose: Scans host environment for required tools and generates lockfiles.

Key Types:

- `Tool_Entry` - Single tool with name, path, hash, version
- `Tool_Registry` - Collection of tools
- `Toolchain_Lockfile` - Complete lockfile data

Key Functions:

- `Initialize_Registry` - Set up builtin tools
- `Resolve_Tool` - Find tool in PATH, compute hash
- `Scan_Toolchain` - Full toolchain scan

3. Config Manager

Files: `build_config.ads/adb` , `config_parser.ads/adb`

Purpose: Manages build configuration and profiles.

Key Types:

- `Build_Profile` - Runtime selection (Auto/Native/Python/Shell)
- `Build_Phase` - Build phase enumeration
- `Configuration` - Complete build config

Key Functions:

- `Initialize_Config` - Create default config
- `Set_Default_Paths` - Set conventional paths
- `Validate_Config` - Validate configuration

4. Dependency Resolver

Files: `dependency_types.ads/adb` , `dependency_resolver.ads/adb`

Purpose: Manages dependency acceptance and resolution.

Key Types:

- `Dependency_Entry` - Single dependency
- `Dependency_Registry` - Collection of dependencies
- `Acceptance_Receipt` - Receipt for accepted dependency

Key Functions:

- `Add_Dependency` - Add dependency to registry
- `Resolve_Dependency` - Resolve single dependency
- `Verify_Hash` / `Verify_Version` - Verification

5. Receipt Manager

Files: `receipt_types.ads/adb` , `receipt_generator.ads/adb`

Purpose: Generates and manages build receipts.

Key Types:

- `Build_Receipt` - Receipt with target, status, epoch
- `Receipt_Registry` - Collection of receipts
- `Input_File_Entry` - Input file with hash

Key Functions:

- `Initialize_Receipt` - Create new receipt

- `Generate_Compilation_Receipt` - Receipt from compilation
- `Finalize_Receipt` - Validate and finalize

6. Build Orchestrator

Files: `build_orchestrator.ads/adb`

Purpose: Coordinates the build pipeline.

Key Types:

- `Build_Result` - Overall build result
- `Build_State` - Build execution state

Key Functions:

- `Run_Build` - Execute full build pipeline
- `Detect_Runtime` - Detect available runtime
- `Execute_Phase` - Execute single phase

Contracts

Preconditions

Procedure	Precondition
<code>Add_Dependency</code>	<code>Reg.Count < Max_Dependencies</code>
<code>Add_Input</code>	<code>Path.Length > 0</code> and <code>Receipt.Input_Count < Max_Input_Files</code>
<code>Add_Tool</code>	<code>Reg.Count < Max_Tools</code>
<code>Parse_Acceptance_Receipt</code>	<code>Receipt_Path.Length > 0</code>

Postconditions

Procedure	Postcondition
<code>Initialize_Registry</code>	<code>Reg.Count >= 2</code>
<code>Initialize_Registry (deps)</code>	<code>Reg.Count = 0</code>
<code>Select_Epoch</code>	<code>Selection.Is_Deterministic</code> or <code>Allow_Current</code>

Statistics

Metric	Value
New Files	14
Specification Files	7
Body Files	7
Lines of Code	~2,200
SPARK_Mode Units	14
Preconditions	12
Postconditions	8

Integration

With Phase 1

Phase 2 components use Phase 1 utilities:

- `Stunir_Strings` - Bounded string types
- `Stunir_Hashes` - Hash types

Build

```
cd core
gprbuild -P stunir_core.gpr -j4
```

Testing

Test files in `core/tests/` :

- `test_epoch.adb` - Epoch manager tests
 - `test_toolchain.adb` - Toolchain scanner tests
 - `test_config.adb` - Config manager tests
 - `test_build.adb` - Build orchestrator tests
-

Phase 2 Status: Complete