

# Minimal + Custom Hooks

*Combining content focus with bespoke roadmaps*

Presentate Team      2026-01-20

Part 1

# Theme Synergy

---

# 1 Theme Synergy

## OUTLINE

1.1 The Best of Both Worlds

1.2 How Hooks Overlap

## SECTION STRUCTURE

# 1 Theme Synergy

## OUTLINE

### **1.1 The Best of Both Worlds**

### 1.2 How Hooks Overlap

## SECTION STRUCTURE

### 1.1.1 Clean Canvas

### 1.1.2 Bespoke Transitions

## 1.1.1 Clean Canvas

This example uses the `minimal` theme, which provides a clean canvas without persistent UI elements (no sidebars, headers, or footers).

## 1.1.2 Bespoke Transitions

We've injected the complex transition slides from the custom-transition example using **Hooks**.

# 1 Theme Synergy

## OUTLINE

1.1 The Best of Both Worlds

**1.2 How Hooks Overlap**

## SECTION STRUCTURE

1.2.1 Default Engine

1.2.2 Manual Override

## 1.2.1 Default Engine

The `minimal` theme would normally use the **Unified Transition Engine** to show a simple roadmap.

## 1.2.2 Manual Override

By providing on-section-change and on-subsection-change functions, you override the engine's default behavior with your own logic.

Part 2

# Configuration Details

---

## 2 Configuration Details

OUTLINE

2.1 Injecting the Logic

2.2 Numbering Propagation

SECTION STRUCTURE

## 2 Configuration Details

OUTLINE

SECTION STRUCTURE

### 2.1 Injecting the Logic

### 2.2 Numbering Propagation

## 2.1 Injecting the Logic

The injection is done via the template parameters:

```
#show: template.with(  
  on-section-change: my-section-transition,  
  on-subsection-change: my-subsection-transition,  
  ...  
)
```

## 2 Configuration Details

OUTLINE

2.1 Injecting the Logic

**2.2 Numbering Propagation**

SECTION STRUCTURE

## 2.2 Numbering Propagation

Even with custom hooks, the global show-heading-numbering and numbering-format options are respected.

Part 3

# Conclusion

---

# 3 Conclusion

OUTLINE

3.1 Summary

SECTION STRUCTURE

# 3 Conclusion

OUTLINE

**3.1 Summary**

SECTION STRUCTURE

## 3.1 Summary

The hook system provides maximum flexibility:

- Use `minimal` for content focus.
- Use custom functions for high-impact transitions.
- Maintain structural consistency via the global configuration.