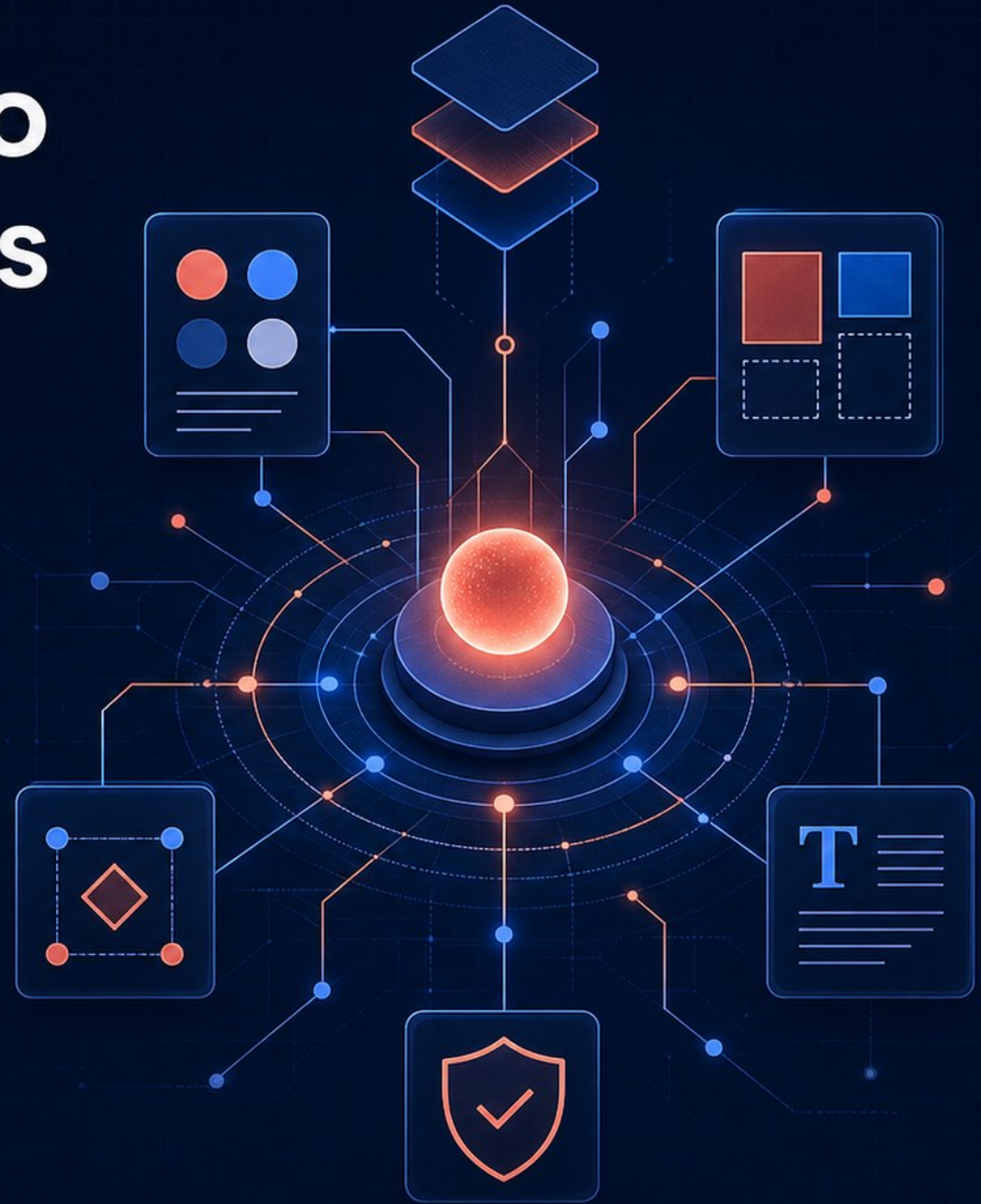


Introduction to Design Systems

- “A design system is a full stack: tokens, components, patterns, docs, and governance—not just a style guide or Figma library”
- “Organizations invest for speed, cost, quality, and brand trust through measurable, compounding reuse”
- “Mature systems cut UI tickets 40–60%, speed up sprints by 33%, and deliver 2.4x faster cycle times”
- “This course covers anatomy, tokens, components, governance, measurement, and your 90-day launch playbook”



The Business Case: Why Design Systems Matter Now



- 84% of teams use design tokens; 74% of Fortune 500 own a design system



- ROI is measured across four pillars: Speed, Cost, Quality, and Brand



- 18-month typical break-even; 47% faster dev time is achievable



- Design systems are now engineering leverage, not just consistency tools

DESIGN SYSTEM IMPACT DASHBOARD REAL-TIME OVERVIEW

● LIVE | 10:24:37 AM



FINANCIAL OUTLOOK



3-YEAR ROI

312%

STRONG RETURN



VALUE
MULTIPLIER
OVER 3 YEARS

Core Anatomy: The Layers of a Modern Design System



- Tokens → Primitives → Components → Patterns → Ecosystem layers form the stack



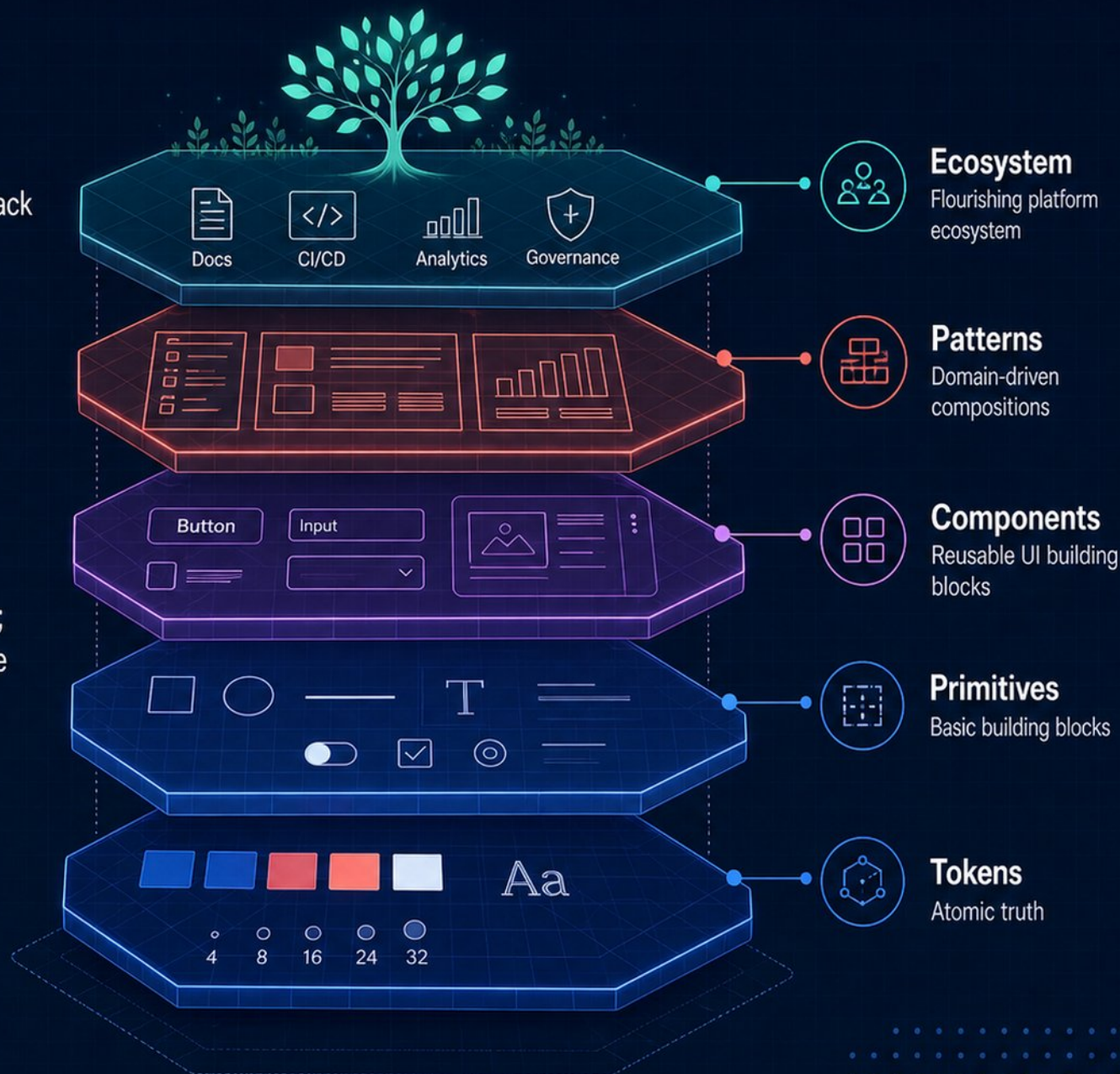
- Tokens provide the single source of truth, bridging design and code



- Components are reusable, accessible UI building blocks; patterns add domain purpose



- Ecosystem includes docs, CI/CD, analytics, and governance—not just a Figma file



Design Tokens: The Foundation Layer

TOKEN-TO-PLATFORM BRIDGE



- Design tokens: named, platform-agnostic variables for all visual decisions



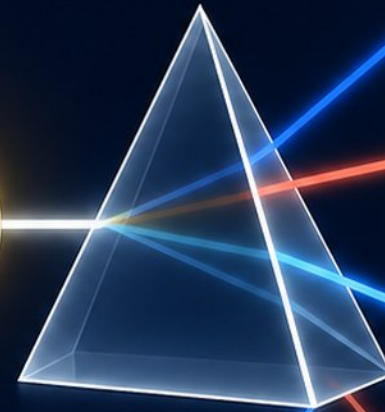
- Three tiers: Primitive (raw values), Semantic (purpose), Component (scoped)



- Bridge design and code across web, iOS, and Android from one source of truth



- W3C DTCG stable spec v2025.10 enables vendor-neutral JSON interchange



```
{ } web (CSS)

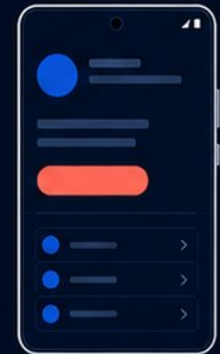
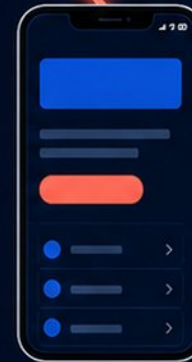
:root {
  --color-primary: #0D6EFD;
  --color-accent: #FF6B4A;
  --radius-md: 8px;
  --space-2: 8px;
}
```

```
🍏 iOS (Swift)

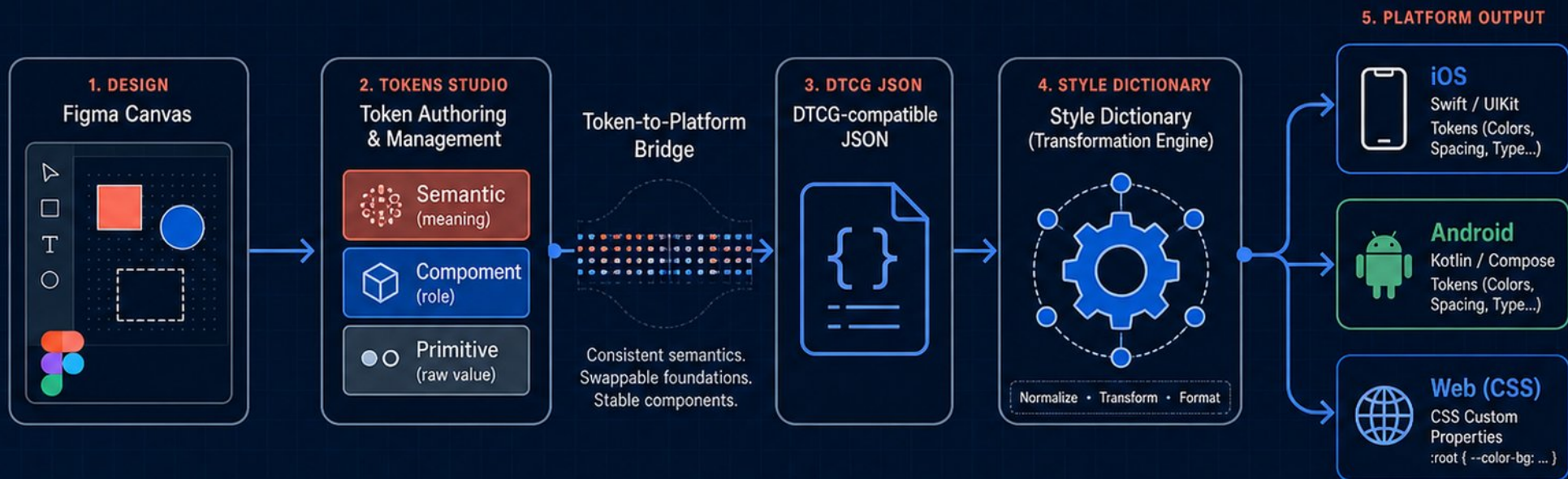
enum ColorTokens {
  static let primary =
    UIColor(hex: "0D6EFD")
  static let accent =
    UIColor(hex: "FF6B4A")
}
```

```
🤖 Android (Kotlin)

object ColorTokens {
  val Primary = Color(0xFF0D6EFD)
  val Accent = Color(0xFFFF6B4A)
}
```



Building a Token Pipeline: From Design to Code



- **Naming conventions:** purpose-first, with primitive, semantic, and component tiers that scale across brands and themes



- **The pipeline:** Figma Variables → Tokens Studio → DTCG-compatible JSON → Style Dictionary → platform output (CSS, iOS, Android)



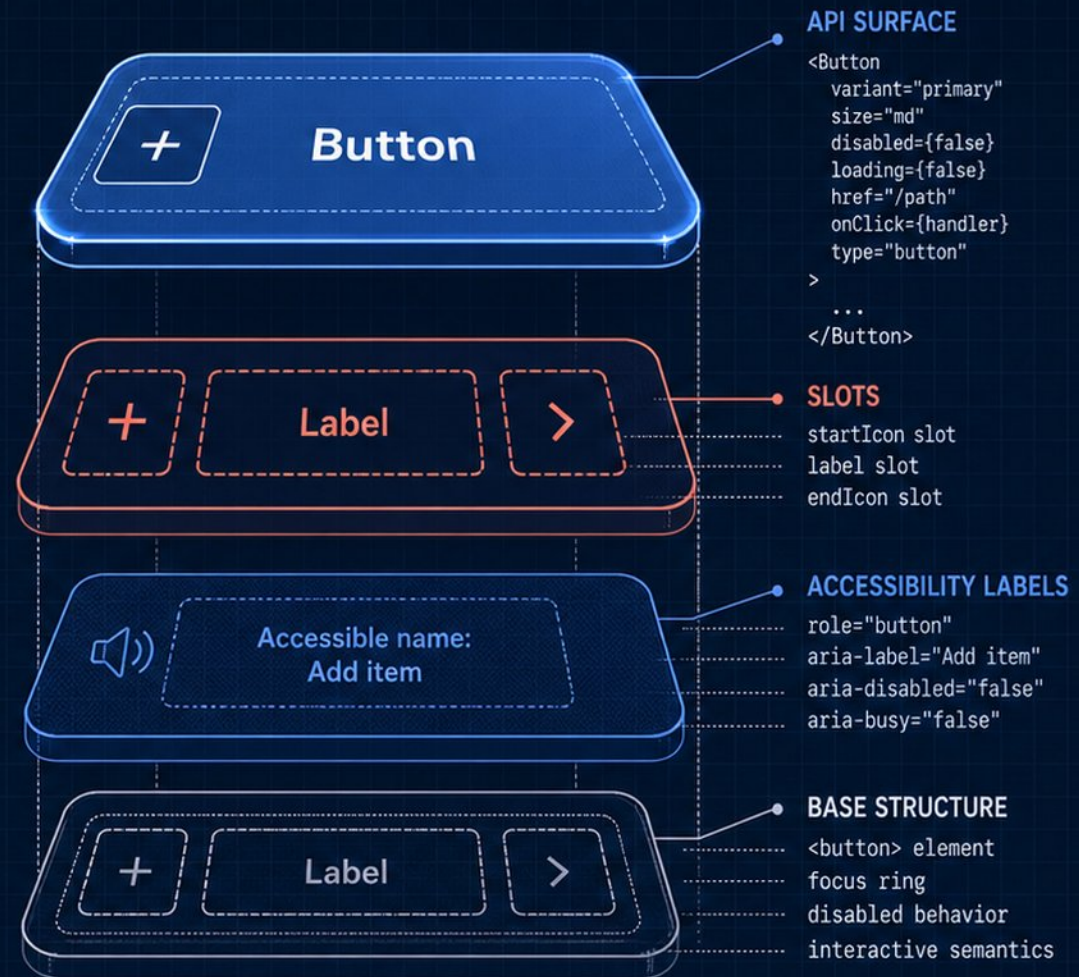
- **Multi-brand theming** and dark mode via semantic token layering—swap primitives, components never change



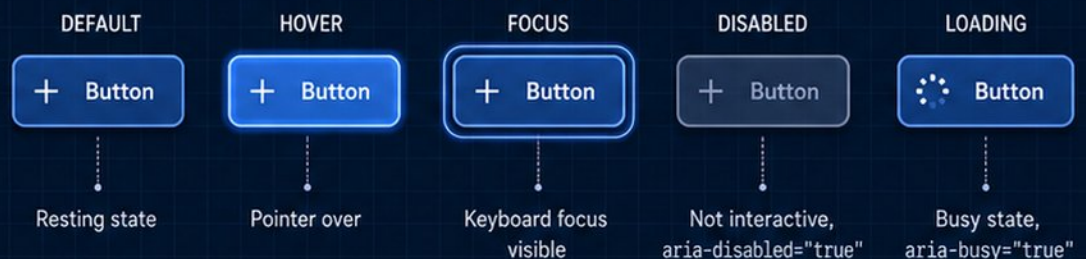
- **Tooling:** Tokens Studio + Style Dictionary + DTCG v2025.10 stable format for vendor-neutral, CI/CD-ready token delivery

Component Library Architecture

- Primitive components (Box, Stack, Grid) centralize layout, backed by tokens
- Composite components (Button, Card, Modal) solve real, recurring product needs
- API design: props, slots, render props, and compound components for flexibility
- Accessibility baked in: WCAG 2.2, ARIA roles, keyboard navigation, focus order
- Versioning, visual regression testing, and npm distribution keep the system stable



STATE VARIANTS



Testing and Quality Assurance for Components

❌ FAILED VISUAL REGRESSION



- ❌ Layout shift
- ❌ Pixel mismatch
- ❌ Contrast fail
- ❌ Alignment off

VS

✅ PASSED VISUAL REGRESSION



- ✅ Layout stable
- ✅ Pixel match
- ✅ Contrast pass
- ✅ Alignment OK



- Unit, interaction, visual regression, and accessibility tests in CI/CD



- Storybook, Chromatic, axe-core, and automated audit tools



- Visual regression catches UI drift and pixel-level bugs before release



- Monitor performance and bundle size for shared libraries

Documentation That Drives Adoption



- **Essential sections:** foundations, components, patterns, and contribution guides



- **Platform comparison:** Storybook, Zeroheight, Supernova, custom sites, and AI-assisted docs

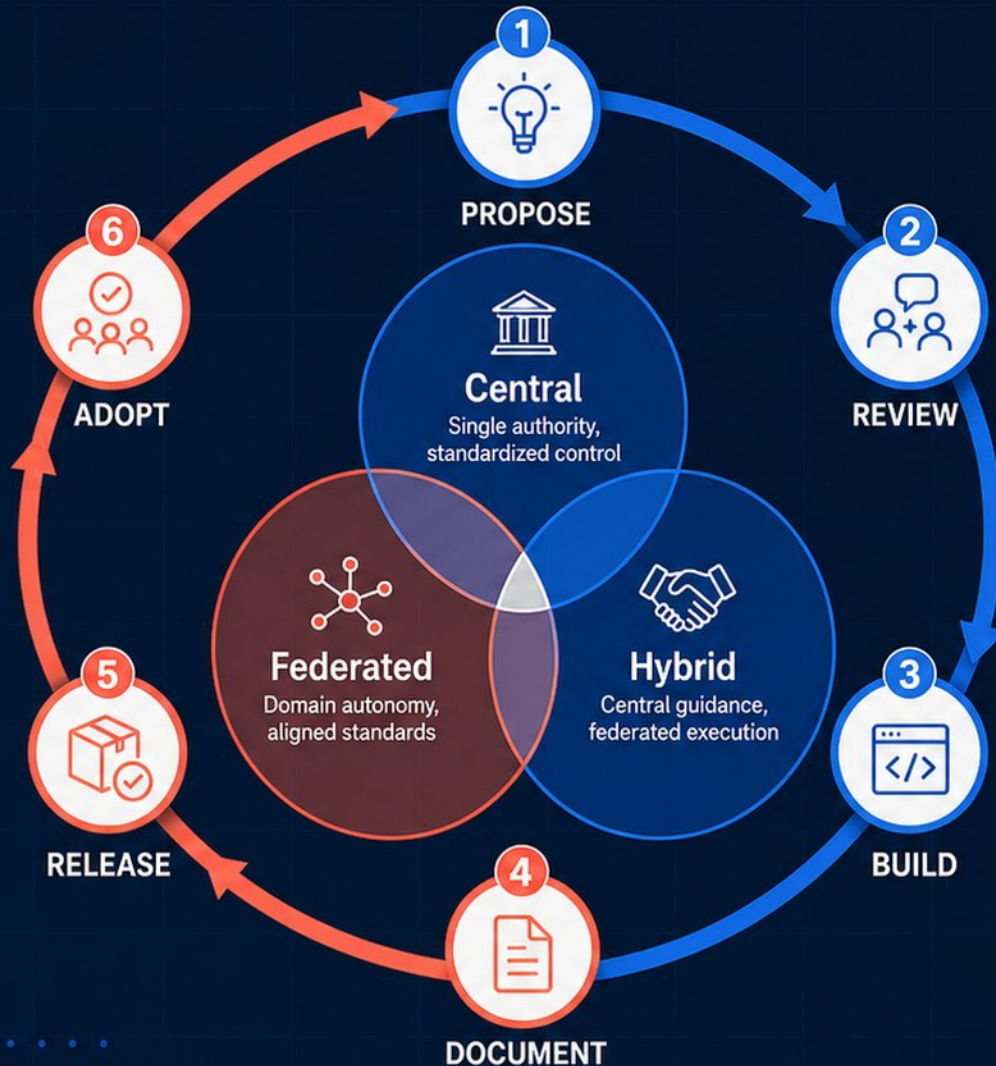


- **Living documentation** through automated publishing, CI/CD integration, and real-time sync



- **Measuring doc health:** coverage, freshness, and feedback loops to prevent drift

Governance Models and Contribution Workflows



- Centralized, federated, and hybrid governance models compared
- Roles: system owners, maintainers, contributors, and consumers
- Contribution lifecycle: propose, review, build, document, release, adopt
- Breaking change management with semantic versioning and migration guides
- Governance is daily practice, not a document—protects coherence at scale

Measuring Design System Success



- **Adoption metrics:** component usage, import tracking, visual coverage analysis



- **Efficiency metrics:** time to first mockup, time to ship, handoff speed



- **Quality metrics:** UI defect reduction, accessibility compliance, consistency scores



- Tie metrics to outcomes budget-holders care about, not internal trivia



- Build dashboards and reporting cadences that resonate with stakeholders



ADOPTION

How widely it's used

72%

Component Usage

1,248

Unique Imports

68%

Visual Coverage



SYSTEM
HEALTH
86%

HEALTHY



EFFICIENCY

How much time we save

2.4 days

Time to First Mockup

11.6 days

Time to Ship

1.8 days

Handoff Speed



QUALITY

How reliable and consistent

-42% ↓

UI Defect Reduction
vs. last 2 quarters



96%

Accessibility Compliance



91%

Consistency Score



Measure what matters. Prove impact. Earn trust.



Lower costs



Faster time to market



Happier teams



Better experiences

Common Pitfalls and How to Avoid Them

- - Start too big or abstract → premature system-ification before real problems exist
- - Figma-code drift: treat both as equal mirrors, not one-way handoffs
- - Neglect dev buy-in → perfect components no one can build
- - Under-resource maintenance: treat the system as a product, not a project



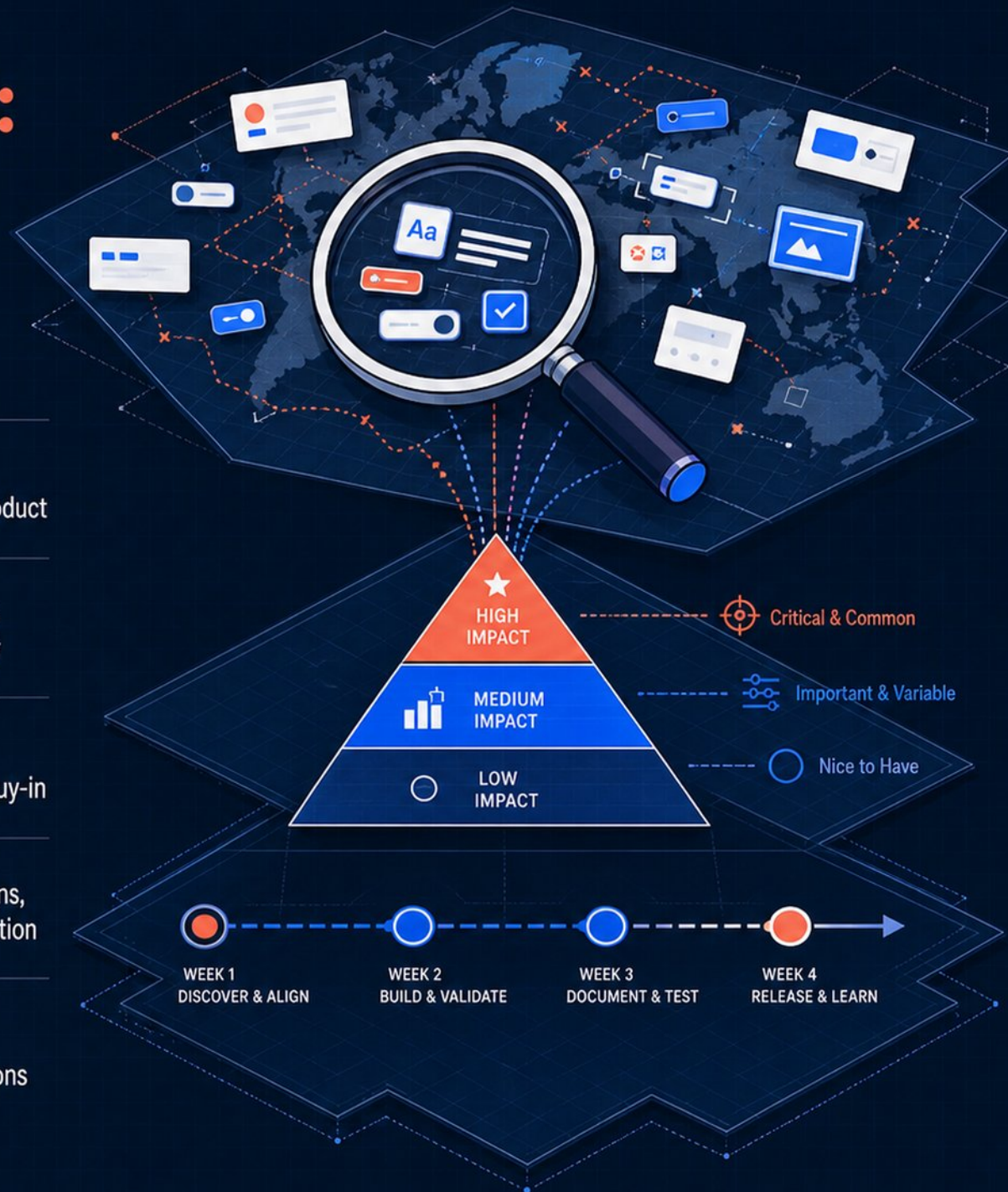
People, Process, and Technology: The Success Triad

- ✓ Build trust and enable autonomy—avoid the "design police" perception
- ✓ Use RFCs, decision logs, and lightweight governance to protect coherence
- ✓ Choose tools that reduce friction, not add overhead
- ✓ Without governance, small deviations silently fracture the system



Getting Started: From Audit to First Release

-  1 Conduct an interface inventory to surface inconsistencies across your product
-  2 Prioritize first tokens and components by frequency, variability, and criticality
-  3 Build a roadmap with a clear business case to secure stakeholder buy-in
-  4 Define the MVP scope: semantic tokens, five core components, and documentation
-  5 Plan a 2–4 week first release that grows from real usage, not assumptions



The 90-Day Playbook: From Zero to a Governed System



- Week-by-week plan:
tokens → core components →
docs → governance →
pilot adoption



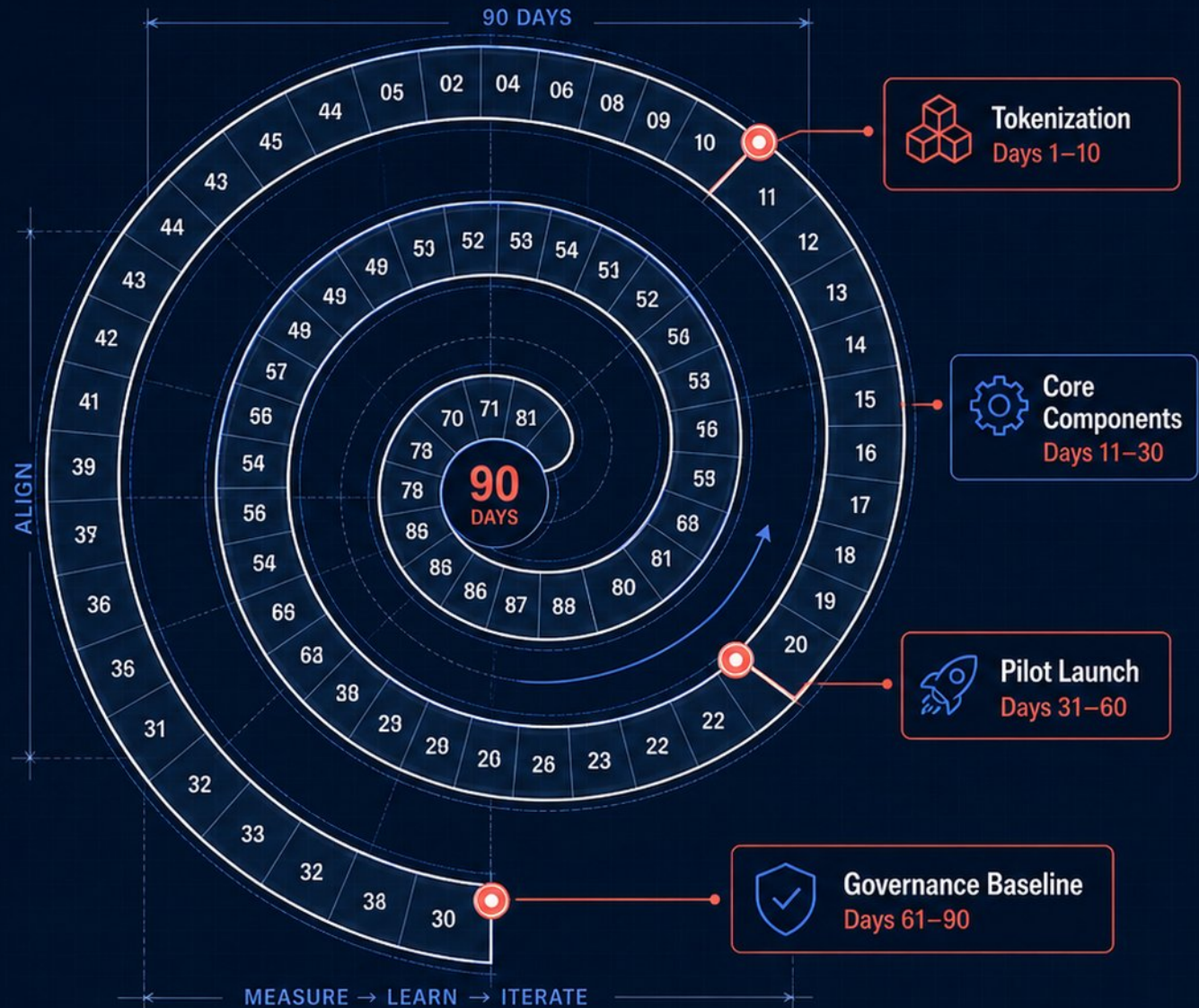
- Minimal viable team:
1 design system lead +
0.5–1.0 FTE engineering



- Launch with one pilot team;
gather structured feedback,
then iterate

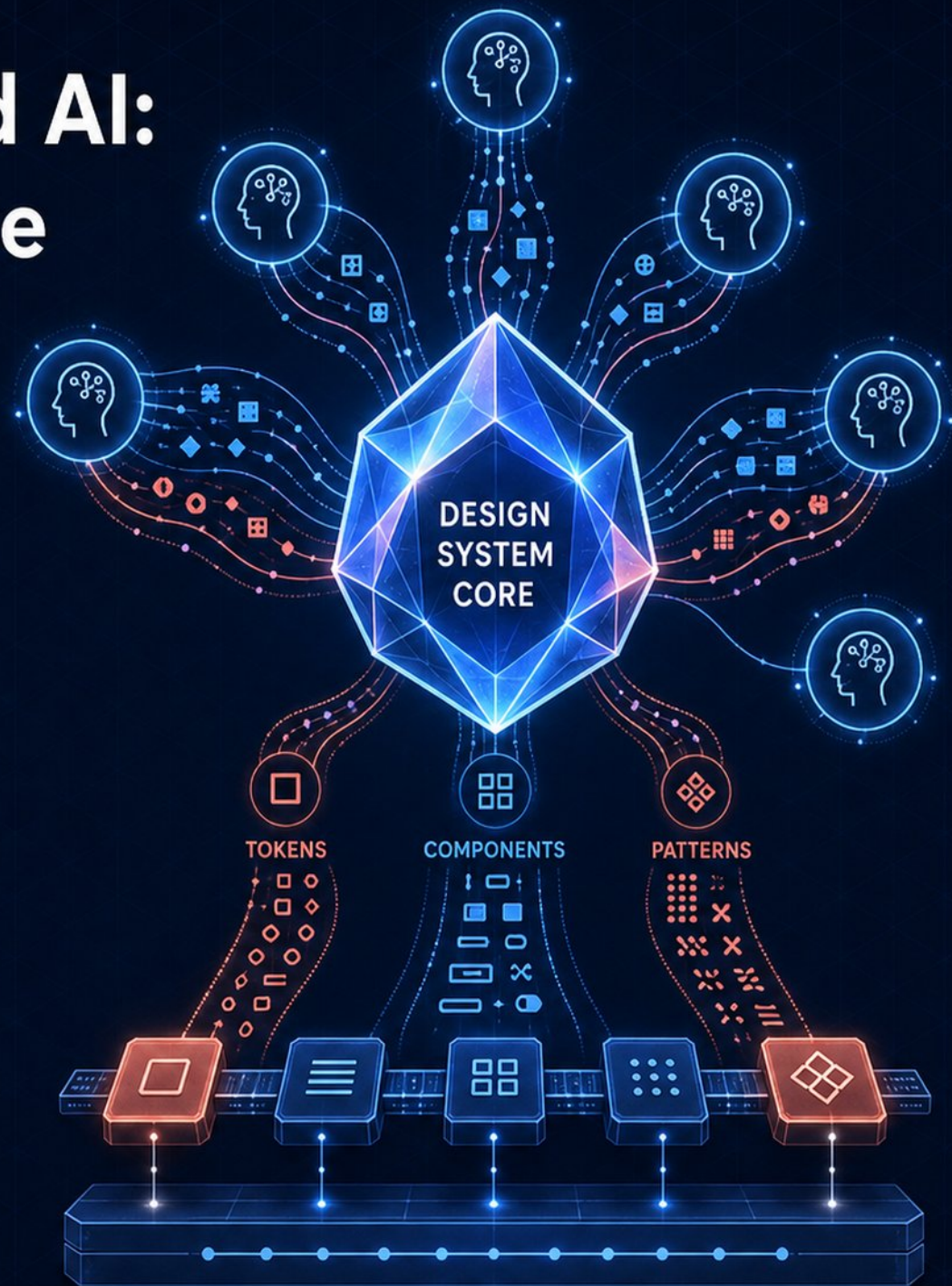


- Baseline metrics before
launch: time-to-ship,
UI defects, adoption signals



Design Systems and AI: The 2026 Landscape

- AI coding tools (Claude Code, Cursor, v0) reward code-first, token-driven systems
- MCP servers expose tokens, components, and patterns to AI agents at edit time
- AI-assisted docs, accessibility annotation, and story generation reduce maintenance
- Only ~10% of teams use AI for docs—the biggest underexploited opportunity in 2026



Key Takeaways and Next Steps

BEFORE: UI CHAOS



PEOPLE

TRIAD
FOR LONG-TERM
SYSTEM
SUCCESS

PROCESS

TECH

GOVERNANCE

Guides • Aligns • Evolves

SUSTAINED IMPACT

AFTER: DESIGN SYSTEM



TOKENS



Aa Aa Aa
Text Heading Display

COMPONENTS

Button Button

Label

☒ Checkbox

Dropdown

PATTERNS

Modal

Data Table

Form

FOUNDATIONS



Color



Typography



Spacing



Radius



Shadow



Motion

DOCUMENTATION



Guidelines



Usage



Do / Don't



Accessibility

ECOSYSTEM



Assets



Packages



Integrations



Community



Recap: core layers from tokens to ecosystem
solve UI chaos at scale



People, process, and tech triad sustains
long-term system success



This week: audit your UI, tokenize a color,
document one component, measure



Curated resources: W3C DTCG spec,
Building Design Systems (2026), DesignSystems.one

PersonWise.

PersonWise • AI digital-human course platform



Scan the code or click anywhere to watch this course live, with voice Q&A

personwise.ai/t/340159d2/public