PSRM Implementation Decisions v1.0

Date: 2026-01-10
Status: DRAFT for review
Scope: Integration of Personal Sigillin Resonance Maps (PSRM) into NeuroProfile v27
Framework: GenesisAeon Feldtheorie / UTAC Charter compliance

Context: From NeuroProfile to PSRM

NeuroProfile v27 (published on Zenodo) provides:

- β-estimation (logistic heuristic)
- σ_Φ-proxy (spectral entropy)
- Gamma-beta coupling (v_RIG proxy)
- Null model falsifiability (ΔAIC)
- Ethics (consent + anonymization)

PSRM extends this with:

- Full CREP metric (Coherence + Resonance + Emergence + Potential)
- Trilayer structure (Signal → Intention → Context)
- BCI calibration protocol (3-phase)
- Hardware abstraction (simulated → prosumer → research)

This document resolves 7 critical design choices before implementation.

Decision 1: CREP Operationalization

Options Considered:

1. **Equal weighting**: CREP = (C + R + E + P) / 4
2. **Adaptive weighting**: Learn weights from pilot data
3. **Hierarchical**: C and R as prerequisites, E and P as amplifiers

DECISION: Equal weighting (Option 1)

Rationale:

- No training data yet to justify adaptive weights
- Aligns with original Living Crystal framework (CREP ≈ 0.84 for conscious-like states)
- Falsifiable: Can be challenged with pilot study results
- Simple to implement and explain

Formula:

```python
def compute_crep(beta, sigma_phi, eeg_data, fs):
    C = coherence(eeg_data)       # Cross-channel phase consistency
    R = gamma_beta_coupling(...)  # From microtubule_resonance.py
    E = min(1.0, beta / 7.4)      # Normalized to biological baseline
    P = exp(-10 * |sigma_phi - 0.0625|)  # Distance from living signature
    return (C + R + E + P) / 4
```

Code Impact:

- NEW: `experiments/.../NeuroProfile/code/crep_calculator.py`
- MODIFY: `neuro_profile_model.py` (add CREP computation to analyze())
- UPDATE: `data/results.json` schema

Validation:

- Bootstrap CI for each component
- AUC comparison: CREP vs individual metrics
- Pilot study: Does CREP > 0.80 predict BCI performance?

Decision 2: Hardware Target Prioritization

Options Considered:

1. **Simulated-only** (current state)
2. **Consumer-first** (Muse, 4ch, $250)
3. **Prosumer-first** (OpenBCI, 16ch, $1000)
4. **Research-grade** (g.tec, 64ch, $10k+)

DECISION: Simulated → Prosumer (Option 1+3)

Rationale:

- **Phase 1 (v10.0)**: Keep synthetic data for MVP development
- **Phase 2 (v10.1)**: Integrate OpenBCI Cyton (8ch) or Daisy (16ch)
- **Skip consumer tier** (4ch too limiting for CREP coherence calculation)
- **Research-grade** only if funding available (v10.2+)

Hardware Profiles (defined in `config/hardware_profiles.yml`):

Tier	Channels	Hz	Expected AUC	CREP Stability	Cost
Simulated	64	1000	0.92	0.87	$0
Prosumer	16	250	0.86	0.81	$600-1000
Research	64+	512+	0.93	0.89	$10k-50k

Code Impact:

- NEW: `config/hardware_profiles.yml`
- NEW: `code/hardware_adapter.py` (abstraction layer for EEG streams)
- MODIFY: `bci_calibrator.py` (graceful degradation for low-channel EEG)

Migration Path:

1. Implement LSL (Lab Streaming Layer) interface
2. Test with OpenBCI synthetic stream
3. User study with prosumer hardware

Decision 3: v_RIG Integration Scope

Options Considered:

1. **Direct measurement** (13.5 MHz magnetometer – not available)
2. **Gamma-beta proxy** (current implementation)
3. **Multi-band cross-frequency coupling** (theta-gamma, alpha-beta)

DECISION: Gamma-beta proxy only (Option 2)

Rationale:

- Current EEG hardware limited to 0.5-100 Hz (Nyquist constraint)
- 13.5 MHz = 135,000× higher than standard EEG bands
- Gamma-beta coupling is **testable proxy** for deeper resonance
- Literature supports gamma-beta as integration signature (IIT)

Implementation (already in `microtubule_resonance.py`):

```python
gamma_power = bandpower(eeg_data, 30, 100, fs)
beta_power = bandpower(eeg_data, 13, 30, fs)
coupling = tanh(gamma_power / (beta_power + 1e-6))
```

Future-proofing:

- Reserve `layer_1_signal.v_rig_direct` field in PSRM schema
- If custom hardware becomes available (v10.5+), add parallel measurement
- Document as "v_RIG proxy (indirect)" in all outputs

Code Impact:

- NO CHANGES (already implemented)
- CLARIFY: Add docstring noting this is a proxy, not direct measurement

---

Decision 4: Trilayer Schema & Sigillin Format

Options Considered:

1. **Custom PSRM schema** (independent of Mandala)
2. **Extend Mandala Sigillin v2.0** (backward compatible)
3. **JSON-only** (no YAML)

DECISION: Extend Mandala Sigillin v2.0 (Option 2)

Rationale:

- Unified Mandala already defines Sigillin architecture
- PSRM is a "neuroscience extension" of the framework
- YAML for human readability (scientists inspect maps)
- JSON API available via `yaml.safe_load()` → `json.dumps()`

Schema Structure:

```yaml
$schema: "https://genesisaeon.org/schemas/neuroprofile/v1"
version: "1.0"
user_id: "<hashed_subject_id>"

layer_1_signal:
  source: "synthetic_1khz_eeg"  # or "openbci_16ch"
  beta: 1.37
  beta_cluster: "sub-critical"
  sigma_phi: 0.94
  gamma_beta_coupling: 0.72
  crep_baseline:
    coherence: 0.82
    resonance: 0.72
    emergence: 0.19
    potential: 0.15
    crep_aggregate: 0.47

layer_2_intention:
  calibrated_intents: []  # Empty until BCI training
  intent_ontology_version: "1.0"
  calibration_status: "pending"

layer_3_context:
  ethics_consent: true
  hardware_tier: "simulated"
  data_retention: "study_duration_only"
  timestamp: "2026-01-10T14:30:00Z"

metadata:
  neuroprofile_version: "v27"
  psrm_mapper_version: "1.0"
  falsifiability_metrics:
    beta_ci: [1.17, 1.58]
    sigma_phi_ci: [0.94, 0.94]
    best_null_model: "constant"
```

Code Impact:

- NEW: `code/psrm_mapper.py` (NeuroProfile → PSRM conversion)
- NEW: `schemas/psrm_sigillin_v1.json` (JSON Schema for validation)
- UPDATE: `data/sigillin_maps/` (new output directory)

Validation:

- JSON Schema validation on all generated maps
- Backward compatibility test with Mandala v0.8.0+

Decision 5: Ethics Enforcement Mode

Options Considered:

1. **Advisory only** (warnings, no blocking)
2. **Soft blocking** (requires manual override)
3. **Hard blocking** (system refuses to proceed)

DECISION: Advisory with audit trail (Option 1+)

Rationale:

- Research context requires flexibility for IRB-approved studies
- Production BCI would use hard blocking (future)
- Audit trail enables post-hoc ethics review
- Aligns with NeuroProfile's consent-first design

Implementation:

```python
class EthicsGuard:
    def check_before_analysis(self, context):
        warnings = []

        if context['location'] == 'public':
            warnings.append("PUBLIC_MODE_RISK: Reduced granularity recommended")

        if context['crep'] < 0.70:
            warnings.append("FRAME_DRIFT: Recalibration advised")

        if not context['consent_granted']:
            raise EthicsViolation("CONSENT_REQUIRED")  # Hard block

        return {"action": "proceed", "warnings": warnings}
```

Logging:

- All warnings logged to `data/ethics_audit.log`
- Format: `[timestamp] [user_hash] [warning_code] [context]`
- Anonymized logs included in study reports

Code Impact:

- NEW: `code/ethics_guard.py`
- MODIFY: `neuro_profile_model.py` (add ethics check in analyze())
- NEW: `data/ethics_audit.log`

Future (v10.2+):

- Physical "kill switch" button for hardware BCI
- Emergency pause protocol (30sec cooldown before resume)

Decision 6: Repository Structure

Options Considered:

1. Keep in `experiments/Phaethon_Geminiden_Bennu/NeuroProfile/`
2. Move to `models/consciousness/neuroprofile/`
3. Split: Core in `models/`, studies in `experiments/`

DECISION: Keep in experiments/, create symlinks (Option 1+)

Rationale:

- NeuroProfile is already published with this path (Zenodo DOI points here)
- Breaking path would invalidate citations
- Symlinks allow logical organization without breaking references

Structure:

```
experiments/Phaethon_Geminiden_Bennu/NeuroProfile/  # Main location
├── code/
│   ├── neuro_profile_model.py
│   ├── crep_calculator.py  # NEW
│   ├── psrm_mapper.py      # NEW
│   └── bci_calibrator.py   # NEW
├── data/
│   ├── sigillin_maps/      # NEW
│   └── ethics_audit.log    # NEW
└── config/
    └── hardware_profiles.yml  # NEW

models/consciousness/  # Symlink for discoverability
└── neuroprofile -> ../../experiments/.../NeuroProfile

api/endpoints/  # Future
└── sigillin_interface.py  # Planned for v10.1
```

Code Impact:

- NO MOVES (preserves DOI references)
- ADD: Symlink creation in setup script
- UPDATE: README.md with new module descriptions

Decision 7: Testing & Validation Strategy

DECISION: 4-tier testing pyramid

1. **Unit Tests** (pytest)
    - Each CREP component independently
    - PSRM mapper with synthetic NeuroProfile
    - Ethics guard edge cases
2. **Integration Tests**
    - NeuroProfile → PSRM → BCI pipeline
    - Hardware adapter with mock LSL stream
3. **Validation Studies** (Jupyter notebooks)
    - CREP vs AUC correlation
    - Synthetic vs real EEG comparison
    - Null model robustness
4. **Pilot Study** (10 subjects, 10 sessions)
    - Primary: PSRM achieves AUC > 0.90
    - Secondary: CREP predicts BCI stability

Code Impact:

- NEW: `tests/test_crep_calculator.py`
- NEW: `tests/test_psrm_mapper.py`
- NEW: `analysis/psrm_validation.ipynb`
- UPDATE: CI pipeline (GitHub Actions)

Migration Timeline

Phase 0: Decision Finalization (Current)

- [ ] Review this document
- [ ] Resolve any conflicting assumptions
- [ ] Get explicit approval on all 7 decisions

Phase 1: Core Implementation (Week 1)

- [ ] Implement `crep_calculator.py`
- [ ] Integrate CREP into `neuro_profile_model.py`
- [ ] Unit tests + CI
- [ ] Update results schema

Phase 2: PSRM Integration (Week 2)

- [ ] Implement `psrm_mapper.py`
- [ ] Create Sigillin schema
- [ ] Generate maps from synthetic data
- [ ] Documentation

Phase 3: BCI Stub (Week 3)

- [ ] Implement `bci_calibrator.py` (no hardware)
- [ ] Hardware profiles YAML
- [ ] Integration tests
- [ ] Ethics guard

Phase 4: Validation (Week 4+)

- [ ] Validation notebook
- [ ] Pilot study protocol
- [ ] arXiv preprint draft
- [ ] Hardware acquisition planning

Open Questions (to be resolved before implementation)

1. **CREP weighting**: Should we collect pilot data first to optimize weights?
2. **Hardware budget**: Is $1k for OpenBCI approved? If not, stay simulated?
3. **Intent ontology**: What's the canonical list of intents for calibration?
4. **Data retention**: How long to keep PSRM maps after study ends?
5. **Publication strategy**: Separate PSRM paper, or extend NeuroProfile v28?

Approval Signatures

Proposer: Claude (AI Assistant)
Reviewer: Johann Benjamin Römer
Date: 2026-01-10
Status: AWAITING REVIEW

Changelog

- v1.0 (2026-01-10): Initial draft based on NeuroProfile v27 + PSRM blueprint