Peer Dependency Upgrade Roadmap

Project: QuantumLeap AI Website
Purpose: Eliminate –legacy-peer-deps requirement
Current Status: Required due to peer dependency conflicts
Target: Clean npm install without flags
Date: November 10, 2025

🎯 Executive Summary

Current Situation:
- npm install requires `--legacy-peer-deps` flag
- Caused by peer dependency conflict in form validation packages
- Simple fix: Upgrade `react-hook-form` from 7.53.0 to 7.55.0+

Impact:
- **Risk Level:** Low (minor version update)
- **Effort:** 15-30 minutes
- **Testing Required:** Form validation on all pages
- **Breaking Changes:** None expected

🔍 Root Cause Analysis

Primary Conflict

```
Dependency Tree:
├─ @hookform/resolvers@5.2.2 (installed)
│  └─ peerDependency: react-hook-form@^7.55.0 (requires 7.55+)
└─ react-hook-form@7.53.0 (installed - TOO OLD)
   └─ Conflict: 7.53.0 < 7.55.0
```

Error Message:

```
npm error ERESOLVE unable to resolve dependency tree
npm error peer react-hook-form@"^7.55.0" from @hookform/resolvers@5.2.2
npm error Found: react-hook-form@7.53.0
```

Why –legacy-peer-deps Works

The `--legacy-peer-deps` flag tells npm to:
1. Ignore peer dependency version conflicts
2. Install packages anyway (legacy npm v6 behavior)
3. Trust that versions are compatible enough

Risk: Technically, 7.53.0 vs 7.55.0 is minor, so it works, but it’s not best practice.

📋 Affected Dependencies

Current Versions

```
{
  "react-hook-form": "7.53.0",         // ❌ Needs upgrade
  "@hookform/resolvers": "^5.2.2"      // ✅ OK (requires rhf 7.55+)
}
```

Forms Using These Dependencies

Based on codebase scan:
1. **Consultation Form** (`/app/consultation/page.tsx`)
2. **Lead Magnet Modal** (`/components/lead-magnet-modal.tsx`)
3. **Business Transformation Calculator** (`/components/calculator/profit-potential-calculator.tsx`)
4. **Automation Calculator** (`/components/calculator/automation-calculator.tsx`)
5. **Breach Check Form** (`/app/background-checks/page.tsx`)
6. **Contact Forms** (various exit-intent modals)

Total Forms to Test: 8-10 forms across 5+ pages

🚀 Upgrade Plan

Phase 1: Upgrade react-hook-form (Immediate)

Steps:

```
cd /home/ubuntu/quantumleap_io/nextjs_space

# Backup current state
git add -A
git commit -m "Backup: Before react-hook-form upgrade"

# Upgrade react-hook-form
npm install react-hook-form@^7.55.0 --legacy-peer-deps

# Test clean install (should work without flag now)
rm -rf node_modules package-lock.json
npm install

# If successful, you can now omit --legacy-peer-deps!
```

Expected Outcome:
- `react-hook-form` upgraded to 7.55.x or 7.56.x
- Peer dependency conflict resolved
- Clean `npm install` without flags

Rollback Plan:

```
# If upgrade causes issues
git reset --hard HEAD~1
npm install --legacy-peer-deps
```

Phase 2: Testing (30-45 minutes)

Critical Test Cases:

1. **Form Submission Test**
   ```
   Pages to test:
      - /consultation (book consultation form)
      - /intelligent-automation/contact (contact form)
      - Exit-intent modals (trigger on SMB page)
      - Calculator lead capture forms
   ```

2. **Validation Test**
   ```
   Test scenarios:
      - Submit empty form → Should show validation errors
      - Submit invalid email → Should reject
      - Submit valid data → Should accept and POST
      - Check error messages display correctly
   ```

3. **TypeScript Compilation**
   ```
   bash
      npm run build
      # Should complete without type errors
   ```

4. **Dev Server Test**
   ```
   bash
      npm run dev
      # Should start without warnings
   ```

Test Checklist:
- [ ] All forms load without errors
- [ ] Form validation works (empty fields rejected)
- [ ] Email validation works
- [ ] Form submission reaches API
- [ ] Success/error messages display
- [ ] No console errors (F12)
- [ ] TypeScript compiles clean
- [ ] Production build succeeds

Phase 3: Documentation Update (5 minutes)

Update these files to remove `--legacy-peer-deps` references:

1. **README.md**
   ```bash
   # Before
   npm install –legacy-peer-deps
   ```

# After
npm install
```

1. **DEVELOPER_HANDOVER.md**
   - Remove note about always using –legacy-peer-deps
   - Update quick commands section

2. **YARN_TO_NPM_MIGRATION.md**
   - Add note that peer dependency issue is resolved
   - Update installation instructions

---

# 📊 Version Comparison

## react-hook-form Changelog

| Version | Release Date | Breaking Changes | Notes |
|---|---|---|---|
| 7.53.0 | Oct 2024 | None | Current version |
| 7.54.0 | Oct 2024 | None | Bug fixes |
| 7.55.0 | Nov 2024 | None | Required by @hook-form/resolvers 5.2.2 |
| 7.56.0 | Nov 2024 | None | Latest stable |

**Recommendation:** Upgrade to `^7.56.0` (latest) for future-proofing.

## API Compatibility

All form hook APIs remain unchanged:
- `useForm()` - ✅ Same signature
- `Controller` - ✅ Same props
- `register()` - ✅ Same behavior
- Validation schemas - ✅ Fully compatible

**Migration Risk:** Extremely low (minor version, no breaking changes)

---

# 🔮 Future-Proofing

## Additional Peer Dependency Checks

Run this command to find other potential conflicts:

```
npm install --dry-run 2>&1 | grep "peer dep"
```
```

Preventive Measures

1. **Pin Major Versions:** Use `^` for minor updates only
2. **Regular Updates:** Monthly dependency review
3. **Test Suite:** Add E2E form tests (Playwright/Cypress)
4. **Dependabot:** Enable for automated security updates

Monitoring

```
# Check for peer dependency warnings
npm install 2>&1 | grep -i "peer\|conflict"

# Audit for security issues
npm audit

# Check for outdated packages
npm outdated
```

🎬 Implementation Timeline

Immediate (Today)

- [x] Document current state
- [x] Create upgrade roadmap
- [ ] Execute Phase 1 (upgrade react-hook-form)
- [ ] Execute Phase 2 (testing)

Short-term (This Week)

- [ ] Remove –legacy-peer-deps from all docs
- [ ] Update team on changes
- [ ] Monitor first few deployments

Long-term (This Month)

- [ ] Implement E2E form testing
- [ ] Enable Dependabot
- [ ] Quarterly dependency review process

📝 Command Reference

Before (with –legacy-peer-deps)

```
npm install --legacy-peer-deps
npm install <package> --legacy-peer-deps
```

After (clean install)

```
npm install
npm install <package>
```

Verification

```
# Should work without errors
rm -rf node_modules package-lock.json
npm install

# Should build successfully
npm run build

# Should show no peer dependency warnings
npm install 2>&1 | grep -i "peer"
```

🐛 Troubleshooting

Issue: Form validation breaks after upgrade

Symptoms: Forms don’t validate or submit
Cause: Schema resolver compatibility
Solution:

```
# Update validation resolvers too
npm install @hookform/resolvers@latest yup@latest zod@latest
```

Issue: TypeScript errors in form components

Symptoms: Type errors in `useForm<FormData>()`
Cause: Type definitions changed
Solution: Update `@types/react-hook-form` (usually auto-installed)

Issue: Build fails after upgrade

Symptoms: `npm run build` fails
Cause: Cached build artifacts
Solution:

```
rm -rf .next node_modules package-lock.json
npm install
npm run build
```

✅ Success Criteria

The upgrade is successful when:

1. **Clean Install Works:**
```
bash
   rm -rf node_modules package-lock.json
   npm install  # No --legacy-peer-deps needed!
   # Output: "added 1333 packages" with NO peer warnings
```

2. **All Forms Function:**
   - Manual testing of all 8-10 forms passes
   - Validation works correctly
   - API submissions succeed

3. **Build Succeeds:**
```
bash
   npm run build
   # Output: ✓ Compiled successfully
```

4. **Documentation Updated:**
   - All references to –legacy-peer-deps removed
   - Installation instructions simplified

5. **Team Notified:**
   - Announcement sent about simplified npm install
   - Documentation shared

📞 Support

If issues arise during upgrade:
1. Check this roadmap’s troubleshooting section
2. Review react-hook-form v7.55 changelog
3. Test in isolated dev environment first
4. Have rollback plan ready (git reset)
5. Escalate to team lead if critical issues

📊 Cost-Benefit Analysis

Current State (–legacy-peer-deps)

Pros:
- Works reliably
- No immediate risk

Cons:
- Confusing for new developers
- Extra flag to remember

- Not following best practices
- Masks potential issues

After Upgrade (clean install)

Pros:
- Standard npm workflow
- Better DX (developer experience)
- Follows npm best practices
- Easier onboarding

Cons:
- Requires 30-minute testing
- Minor risk during upgrade

Recommendation: Upgrade ASAP. Benefits far outweigh minimal risk.

🎯 Next Actions

1. **Execute Upgrade:**
   ```
   bash
      cd /home/ubuntu/quantumleap_io/nextjs_space
      npm install react-hook-form@^7.56.0 --legacy-peer-deps
      rm -rf node_modules package-lock.json
      npm install  # Test clean install
   ```

2. **Run Tests:**
   - Manual form testing (all pages)
   - Build verification
   - Dev server verification

3. **Update Docs:**
   - Remove –legacy-peer-deps references
   - Commit changes

4. **Deploy:**
   - Push to staging
   - Test live forms
   - Deploy to production

Created By: DeepAgent (Abacus.AI)
Priority: High (improves DX, low risk)
Estimated Time: 30-45 minutes
Related Docs: YARN_TO_NPM_MIGRATION.md