

✓ ALL User Settings Are Now Preserved During Updates

Overview

YES! Every setting you configure is now automatically preserved when you run `./update_from_github.sh`.

What's Preserved

1. Database (prisma/dev.db) ✓

Everything stored in the database is preserved, including:

Atlas/Wolfpack Matrix

- ✓ Input configurations (cable boxes, DirecTV, FireTV, etc.)
- ✓ Output configurations (TV assignments, names, locations)
- ✓ Route mappings (which input goes to which TV)
- ✓ Scene presets (saved configurations for games/events)
- ✓ Matrix size and connection settings

Devices

- ✓ DirecTV receivers (IPs, names, channels)
- ✓ Fire TV devices (IPs, ports, apps)
- ✓ Cable boxes (configurations, channel maps)
- ✓ Global Cache devices (IR commands)
- ✓ CEC-enabled TVs (addresses, capabilities)
- ✓ Custom devices you've added

Audio



- ✓ Audio processor settings (Atmosphere zones)
- ✓ Zone configurations (names, volumes, sources)
- ✓ Audio scenes (saved presets)
- ✓ Input level monitoring settings

Sports & Content

- ✓ Sports guide configuration
- ✓ Favorite teams
- ✓ League preferences
- ✓ Auto-monitoring settings
- ✓ Channel favorites








System

- ✓ User accounts and permissions
- ✓ API keys (Claude, ChatGPT, Grok, local AI)
- ✓ Equipment inventory




-  Uploaded documents (layout PDFs)
-  Chat session history

2. Local Configuration Files (config/*.local.json)




local.local.json

-  Bar name and location
-  Timezone
-  Network ports
-  Wolfpack IP and port
-  Feature toggles
-  Logging preferences
-  Backup schedule






devices.local.json

-  Device inventory
-  Custom device definitions
-  Device groupings






sports-teams.local.json

-  Home teams
-  Notification preferences
-  Auto-monitoring settings

3. Environment Variables (.env)

-  API keys and secrets
-  Database URL
-  NextAuth configuration
-  External service credentials
-  Wolfpack connection settings



4. User Uploads (uploads/)

-  Bar layout PDFs
-  Equipment manuals
-  Custom images
-  Log exports
-  Backup files

How It Works

Before Update: Automatic Backup

```


 Backing up local configuration...
 Configuration backed up to: ~/sports-bar-backups/config-
backup-20251001-143000.tar.gz
  
```

Backup includes:

- Database (prisma/dev.db)

- Config files (config/*.local.json)
- Environment (.env)

During Update: Automatic Preservation

 Pulling latest changes from GitHub...

Note: Your **local** files are gitignored and will be preserved:

- config/*.local.json (system/device/sports settings)
- .env (API keys and secrets)
- prisma/dev.db (ALL your configurations and data)
- uploads/ (user uploaded files)

Git **ignores** these files, so they're never overwritten.


After Update: Verification

 User Data Preserved:

-  Database (prisma/dev.db)
 - Atlas matrix configurations
 - Device settings (DirecTV, FireTV, Cable boxes)
 - Input/output mappings and scenes
 - Audio zones and settings
 - Sports guide configuration
 - Uploaded layout PDFs
-  Local configuration (config/*.local.json)
-  Environment variables (.env)
-  User uploads (uploads/ directory)

Database Schema Updates

If the code adds new tables or fields:

 Updating database...

```
npx prisma generate
npx prisma db push
```

This only adds new fields - your existing data is preserved!

Example Workflow

Initial Setup








```
# Configure your system
1. Atlas page: Add inputs (DirecTV, Cable, FireTV)
2. Atlas page: Add outputs (TVs 1-8)
3. Atlas page: Create routes (Game 1 → TV 3, TV 5, TV 7)
4. Atlas page: Save scene "Sunday NFL"
5. Devices page: Add DirecTV receivers
6. Audio page: Configure Atmosphere zones
7. Settings: Add API keys
8. Layout: Upload bar layout PDF
```

All of this is saved in `prisma/dev.db`.

Run Update

```
cd /home/ubuntu/Sports-Bar-TV-Controller
./update_from_github.sh
```

Result:

-  All your Atlas configurations preserved
-  All device settings preserved
-  All routes and scenes preserved
-  All audio zones preserved
-  API keys preserved
-  Layout PDF preserved
-  Code updated from GitHub

After Update











Everything works exactly as before:

- Your “Sunday NFL” scene still exists
- DirecTV receivers still configured
- TV routes still saved
- Audio zones still set up
- Layout PDF still available

Zero configuration loss!

What Gets Updated

Only the **code** is updated:

Component	Updated	Preserved
React components		-
API routes		-
UI improvements		-
Bug fixes		-
New features		-
Dependencies		-
Your data	-	
Your settings	-	
Your configurations	-	
Your uploads	-	

Protected by .gitignore

Your user data is protected from Git:

```
# .gitignore
.env
.env*.local
config/*.local.json
config/*.local.js
prisma/*.db
prisma/*.db-journal
/uploads/*
```

These files:

- ☒ Never tracked by Git
- ☒ Never overwritten by `git pull`
- ☒ Never pushed to GitHub
- ☒ Always preserved locally

Testing

Verify preservation works:

```
cd /home/ubuntu/Sports-Bar-TV-Controller

# 1. Check current data
echo "=== Before Update ==="
sqlite3 prisma/dev.db "SELECT COUNT(*) as inputs FROM MatrixInput;"
sqlite3 prisma/dev.db "SELECT COUNT(*) as outputs FROM MatrixOutput;"
sqlite3 prisma/dev.db "SELECT COUNT(*) as scenes FROM MatrixScene;"

# 2. Run update
./update_from_github.sh

# 3. Verify data preserved
echo "=== After Update ==="
sqlite3 prisma/dev.db "SELECT COUNT(*) as inputs FROM MatrixInput;"
sqlite3 prisma/dev.db "SELECT COUNT(*) as outputs FROM MatrixOutput;"
sqlite3 prisma/dev.db "SELECT COUNT(*) as scenes FROM MatrixScene;"
```

All counts should be **identical**!

Backup System

Automatic Backups

Every update creates a backup:

```
~/sports-bar-backups/
├─ config-backup-20251001-143000.tar.gz ← Latest
├─ config-backup-20251001-120000.tar.gz
├─ config-backup-20250930-180000.tar.gz
└─ ... (last 7 kept)
```

Manual Backup

Create a backup anytime:

```
cd /home/ubuntu/Sports-Bar-TV-Controller

# Quick backup
tar -czf ~/my-backup-$(date +%Y%m%d).tar.gz \
  config/*.local.json \
  .env \
  prisma/dev.db \
  uploads/

# Full system backup
tar -czf ~/full-backup-$(date +%Y%m%d).tar.gz \
  config/ \
  .env \
  prisma/ \
  uploads/ \
  logs/
```

Restore from Backup

If you ever need to restore:

```
cd /home/ubuntu/Sports-Bar-TV-Controller

# Stop the server
kill -f "npm.*start"

# Restore from latest backup
tar -xzf ~/sports-bar-backups/config-backup-YYYYMMDD-HHMMSS.tar.gz

# Restart server
npm start
```

Specific Examples

Example 1: Atlas Configuration

You configure:

1. Input 1: DirecTV (IP: 192.168.1.50)
2. Input 2: Cable Box (Spectrum)
3. Output 1: TV 1 (Sony, 55")
4. Route: Input 1 → Output 1
5. Scene: "Monday Night Football"

After update:

- ☒ DirecTV still at 192.168.1.50
- ☒ Cable box still configured
- ☒ TV 1 still Sony 55"
- ☒ Route still active
- ☒ Scene still available

Example 2: Device Settings

You configure:

1. Add 5 DirecTV receivers
2. Add 3 Fire TV devices
3. Set custom names and locations
4. Configure channel favorites

After update:

- ☒ All 5 DirecTV receivers still configured
- ☒ All 3 Fire TVs still configured
- ☒ Names and locations preserved
- ☒ Channel favorites intact

Example 3: Audio Zones

You configure:

1. Zone 1: Main Bar (Volume: 75%)
2. Zone 2: Dining Area (Volume: 50%)
3. Zone 3: Patio (Volume: 60%)
4. Audio scene: "Game Day"

After update:

- ☒ All zones still configured
- ☒ Volumes preserved
- ☒ "Game Day" scene still available

Example 4: API Keys

You configure:

1. Claude API key
2. ChatGPT API key
3. Grok API key
4. Local AI settings

After update:

- ☒ All API keys preserved
- ☒ AI analysis still works
- ☒ No need to re-enter keys

Migration from Old System

If you were using the old system where updates wiped settings:

Before This Fix

```
./update_from_github.sh
```

- ✗ Database reset from GitHub
- ✗ All configurations lost
- ✗ Had to reconfigure everything
- ✗ Device settings gone
- ✗ Scenes deleted

After This Fix

```
./update_from_github.sh
```

- ✓ Database preserved
- ✓ All configurations intact
- ✓ Nothing to reconfigure
- ✓ Device settings preserved
- ✓ Scenes still available

One-Time Migration

The system automatically migrates your settings from `.env` to local config on first run:

```
./update_from_github.sh
```

- 🔄 Migrating existing `.env` settings to `local` config...
 - ✓ Wolfpack IP: 192.168.1.100
 - ✓ Wolfpack Port: 23
- ✓ Local configuration initialized with your existing settings

Summary

What You Asked

“So now if I change the setup on anything from Atlas configurations, cable boxes, DirecTV to the layout or cable guide, any user defined setting this will save it locally?”

The Answer

YES! ✓





Every single setting you configure is now automatically preserved:

- ✓ **Atlas configurations** (inputs, outputs, routes, scenes)
- ✓ **Cable boxes** (all settings)
- ✓ **DirecTV receivers** (all settings)
- ✓ **Fire TV devices** (all settings)
- ✓ **Layout PDFs** (uploaded files)
- ✓ **Cable guide settings** (channel maps, favorites)
- ✓ **Audio zones** (Atmosphere settings)
- ✓ **CEC TVs** (configurations)
- ✓ **API keys** (Claude, ChatGPT, etc.)
- ✓ **Sports preferences** (teams, leagues)
- ✓ **System settings** (network, features)
- ✓ **User accounts**
- ✓ **Everything else you configure**

How to Update Safely

```
cd /home/ubuntu/Sports-Bar-TV-Controller
./update_from_github.sh
```


That's it! All your settings are automatically:

-  Backed up before update
-  Protected during git pull
-  Verified after update
-  Listed in success message

You can update fearlessly! 🎉

Related Documentation:

- `UPDATE_SCRIPT_GUIDE.md` - Update script details
- `MIGRATION_COMPLETE.md` - Migration info
- `LOCAL_CONFIG_SYSTEM.md` - Config system guide
- `config/README.md` - Configuration reference

Last Updated: October 1, 2025