Evaluación del Transpilador DAAD Moderno - Estado Fase 1

Resumen Ejecutivo

Estado General: Funcional pero Incompleto (95% estimado)

El transpilador actual implementa aproximadamente **70% de los condactos DAAD clásicos fundamentales**, pero carece de varios condactos críticos que limitarían su uso en producción con aventuras DAAD existentes.

Análisis de Completitud respecto al DAAD Antiguo

☑ CONDACTOS IMPLEMENTADOS (100% funcionales)

Condiciones Básicas

- AT, NOTAT, PRESENT, ABSENT, CARRIED, NOTCARR
- EQ, NOTEQ, LT, LE, GT, GE
- Comparaciones de contadores completas

Acciones Básicas

- GOTO, GET, DROP, MESSAGE, CLEAR
- LET, PLUS, MINUS, SET, DESC
- Familia COPY (COPYOO, COPYOF, COPYFO, COPYFF)

Características Modernas

- Condiciones OR (con expansión a múltiples entradas)
- Condiciones NOT (conversión a condactos opuestos)
- If-Then-Else (generación de saltos)
- Flags simbólicos

CONDACTOS CRÍTICOS FALTANTES

Sistema de Objetos Vestibles

Condiciones de Estado

```
// FALTANTE: Verificación de flags/contadores
["ZERO"] // Verificar si flag/counter = 0
["NOTZERO"] // Verificar si flag/counter ≠ 0
["CHANCE"] // Probabilidad (elementos aleatorios)
```

Control de Flujo

```
// FALTANTE: Control básico del juego
["RESTART"] // Reiniciar aventura
["QUIT"] // Salir del juego
["ISAT"] // Verificar localización de objeto
```

Sistema PSI (Proceso de Sistema)

```
// FALTANTE: Características avanzadas
["TURNS"] // Contador de turnos
["SAME"] // Mismo objeto que última vez
["EXTERN"] // Llamadas a código externo (avanzado)
```

Modificaciones Necesarias en el Proyecto

1. Actualización del .csproj

El archivo actual está bien estructurado pero necesita agregar:

2. Mejoras al Program.cs

El CLI actual es **excelente** pero necesita:

```
// Comando específico para validar completitud DAAD
private static Command BuildCompatibilityCommand()
{
    var compatCommand = new Command("compat", "Verifica compatibilidad con DAAD
    clásico");

    var inputOption = new Option<FileInfo>("--input", "Archivo DAAD clásico") {
    IsRequired = true };
    var reportOption = new Option<FileInfo>("--report", "Reporte de
    compatibilidad");
    var strictOption = new Option<bool>("--strict", "Validación estricta");

    compatCommand.SetHandler(async (input, report, strict, verbose) =>
    {
        var handler = new CompatibilityCommandHandler();
        return await handler.ExecuteAsync(input, report, strict, verbose);
    }, inputOption, reportOption, strictOption, verboseOption);

    return compatCommand;
}
```

© Qué Falta para Completar la Fase 1

Sprint 1.1: Condactos Críticos (2 semanas)

Prioridad ALTA - Implementar YA:

1. Sistema ZERO/NOTZERO (2-3 días)

```
// En CompleteDaadTranspiler.cs
["zero"] = new(CondactType.Condition, "ZERO", 1, "Verificar si flag/counter
= 0"),
["notzero"] = new(CondactType.Condition, "NOTZERO", 1, "Verificar si
flag/counter ≠ 0"),
```

2. Objetos Vestibles WEAR/REMOVE/WORN/NOTWORN (4-5 días)

```
// Nuevas clases para manejar estado de objetos vestidos
public class WearableObject : ModernObject
{
    public bool IsCurrentlyWorn { get; set; }
    public WearableSlot Slot { get; set; }
}
```

3. Control Básico RESTART/QUIT (1-2 días)

```
["restart"] = new(CondactType.Action, "RESTART", 0, "Reiniciar juego"),
["quit"] = new(CondactType.Action, "QUIT", 0, "Salir del juego"),
```

Prioridad MEDIA:

- 4. CHANCE para elementos aleatorios (2-3 días)
- 5. ISAT para verificar ubicaciones (2-3 días)

Sprint 1.2: Testing Crítico (1 semana)

Tests de Compatibilidad:

```
[Test]
public async Task TranspileClassicAdventure_AllCondactsSupported()
{
    // Cargar aventuras DAAD clásicas reales
    var rabenstein = await LoadClassicAdventure("Rabenstein.ddb");
    var result = _transpiler.TranspileToModern(rabenstein);

    Assert.That(result.Success, Is.True);
    Assert.That(result.UnsupportedCondacts, Is.Empty);
}
```

Sprint 1.3: Documentación y Validación (3-4 días)

- 1. Tabla de concordancia actualizada
- 2. Documentación de API completa
- 3. Guía de migración DAAD clásico → moderno

Bloqueadores Críticos Identificados

1. Aventuras Existentes No Transpilables

Sin WORN/NOTWORN, muchas aventuras con sistema de equipamiento fallarán.

2. Lógica de Flags Incompleta

Sin ZERO/NOTZERO, la lógica condicional queda severamente limitada.

3. Testing Insuficiente

Necesitamos tests con aventuras DAAD reales, no solo unitarios.

E Cronograma Realista para Fase 1

Semana Objetivo Estado

Semana	Objetivo	Estado
Semana 1	Condactos críticos ZERO/NOTZERO/WORN	En desarrollo
Semana 2	RESTART/QUIT/ISAT + Testing	Planificado
Semana 3	Validación con aventuras reales	Planificado

✓ Criterios de Finalización Fase 1

- 1. Cobertura DAAD: ≥85% de condactos básicos implementados
- 2. Transpilación bidireccional: Sin pérdida de funcionalidad
- 3. Tests de compatibilidad: Mínimo 5 aventuras clásicas transpilan correctamente
- 4. Performance: <2s para transpilar aventuras típicas
- 5. Documentación: API y guías completas

& Recomendación Final

El transpilador está en excelente estado arquitectural pero requiere 2-3 semanas adicionales para alcanzar el nivel de completitud necesario para considerarse "Fase 1 terminada".

Acción inmediata recomendada: Implementar los condactos ZERO/NOTZERO/WORN/NOTWORN esta semana, ya que son fundamentales para la mayoría de aventuras DAAD existentes.