

## 9 פרוטוקול נתוני הליגה

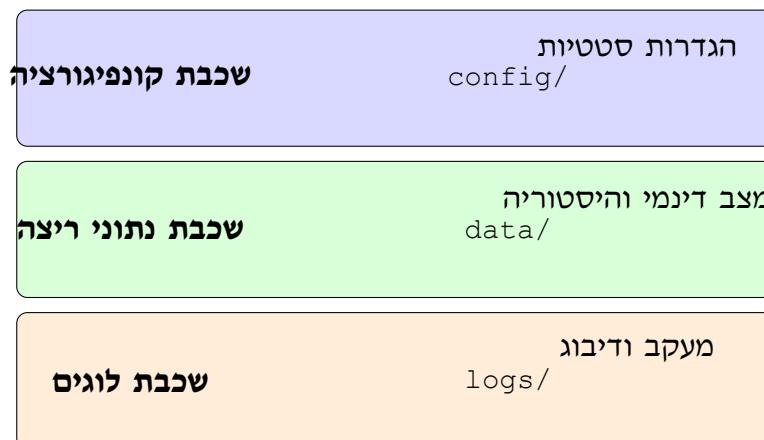
### 9.1 מבוא: הקוד הגנטי של חברת הסוכנים

כasher anu bonyim chibra shel socnims ottonomiyim - shaknim, shoftim v'manhali ligah - anu lema'aseh yozrim terbotot digitalit chadsha. cmu b'khol chibra anoushit, g'm ca'an n'drashim shelosha y'sodot kritiyim:

1. **חוקים משותפים** - ה프וטוקול שהגדנו בפרקים הקודמים.
2. **זיכרון קולקטיבי** - היכולת לשמר ולשחזר מידע היסטורי.
3. **קוד גנטי** - הקונפיגורציה שמנדרה את ה-DNA של כל סוכן.

Nespa' zeh matayr at "basis ha'ntoniim ul kovchi JSON" - architekturat shelosh shabotot she'mafsharat la'meracat l'zmo'ah l'kena' midah shel alpi socnims v'ligot.

### 9.2 ארכיטקטורת שלוש השכבות



#### 9.2.1 עקרונות מנהים

כל kovch b'meracat umid be'ukronot ha'ba'im:

- **מזהה ייחודי (id)**: كل אובייקט ראשי מקבל מזהה חד-ערכי.
- **גרסת סכמה (schema\_version)**: מאפשר מיגרציות עתידית.
- **חותמת זמן (last\_updated)**: בפורמט UTC/ISO-8601.
- **תאימות לפרוטוקול**: כל השדות תואמים ל-2.v.league.

### 9.3 שכבת הקונפיגורציה – config/

shcbe zo mchila at ha'kode' ha'geneti' shel meracat - hagdrot statiyot shnkratot be'uliyot ha'socnims.

### 9.3.1 קובץ מערכת גלובלי - config/system.json

- **מטרה:** פרמטרים גלובליים לכל המערכת.
  - **משתמשים:** כל הסוכנים, Orchestrator עליון.
  - **מיקום:** SHARED/config/system.json
- קובץ זה מגדיר את ערכי בירית המוחדר עבורה:
- הגדרות רשת (network) – פורטים וכתובות.
  - הגדרות אבטחה (security) – טוקנים ו-TTL.
  - זמני המתנה (timeouts) – תואמים להגדרות ה프וטוקול בפרק 2.
  - מדיניות ניסיון חזר (retry\_policy) – תואמת להגדרות הפרוטוקול.

#### דוגמה: מבנה system.json (קטע)

```
{  
    "schema_version": "1.0.0",  
    "system_id": "league_system_prod",  
    "protocol_version": "league.v2",  
    "timeouts": {  
        "move_timeout_sec": 30,  
        "generic_response_timeout_sec": 10  
    },  
    "retry_policy": {  
        "max_retries": 3,  
        "backoff_strategy": "exponential"  
    }  
}
```

### 9.3.2 רישום סוכנים - config/agents/agents\_config.json

- **מטרה:** ניהול מרכזי של אלף סוכנים.
  - **משתמשים:** מנהל הליגה, כלי Deployment.
  - **מיקום:** SHARED/config/agents/agents\_config.json
- קובץ זה מכיל את "ספר האזרכיס" של חברת הסוכנים:
- league\_manager – פרטி מנהל הליגה.
  - referees[] – רשימת כל השופטים הרשומים.
  - players[] – רשימת כל השחקנים הרשומים.

### 9.3.3 קונFIGורציית ליגה - config/leagues/<league\_id>.json

- מטרה: הגדרות ספציפיות לliga.
- משתמשים: מנהל הליגה, שופטים.
- מיקום: SHARED/config/leagues/league\_2025\_even\_odd.json
- כל ליגה היא "מדינה" עצמאית עם חוקים משלها:

#### דוגמה: קונFIGורציה ליגה (קטע)

```
{  
    "league_id": "league_2025_even_odd",  
    "game_type": "even_odd",  
    "status": "ACTIVE",  
    "scoring": {  
        "win_points": 3,  
        "draw_points": 1,  
        "loss_points": 0  
    },  
    "participants": {  
        "min_players": 2,  
        "max_players": 10000  
    }  
}
```

### 9.3.4 רישום סוגי משחקים - config/games/games\_registry.json

- מטרה: רישום כל סוגי המשחקים הנתמכים.
- משתמשים: שופטים (לטינית מודול חוקים), מנהל ליגה.
- מיקום: SHARED/config/games/games\_registry.json
- המערכת תומכת בסוגי משחקים רבים. כל משחק מגדיר:
  - game\_type – מזאה ייחודי.
  - מודול החוקים לטעינה.
  - rules\_module
  - max\_round\_time\_sec – זמן מקסימלי לסיוב.

### 9.3.5 ברירות מחדל לסוכנים - config/defaults/

- מטרה: ערכי ברירת מחדל לפי סוג סוכן.
- קבועים: player.json, referee.json
- מיקום: SHARED/config/defaults/
- קבצים אלה מאפשרים לסוכן חדש להתחיל לפעול עם הגדרות סבירות מוביל להגדר כל פרמטר בנפרד.

## 9.4 שכבת נתונים הריצה - data/

אם שכבת הkonfiguraciah היא ה"קוד המקורי", שכבת נתונים הריצה היא ה"איךرون ההיסטורי" של החברה. כאן נשמרים כל האירועים שקרוים במערכת.

### 9.4.1 טבלת דירוג - data/leagues/<league\_id>/standings.json

- מטרה: מצב הדירוג העדכני של הליגה.

- מערכן: מנהל הליגה (אחרי MATCH\_RESULT\_REPORT).

- מיקום: SHARED/data/leagues/league\_2025\_even\_odd/standings.json

#### דוגמה: מבנה טבלת דירוג

```
{  
    "schema_version": "1.0.0",  
    "league_id": "league_2025_even_odd",  
    "version": 12,  
    "rounds_completed": 3,  
    "standings": [  
        {  
            "rank": 1,  
            "player_id": "P01",  
            "display_name": "Agent_Alpha",  
            "wins": 4, "draws": 1, "losses": 1,  
            "points": 13  
        }  
    ]  
}
```

### 9.4.2 היסטוריה מחזוריים - data/leagues/<league\_id>/rounds.json

- מטרה: תיעוד כל המחזוריים שהתקיימו.

- מערכן: מנהל הליגה (אחרי ROUND\_COMPLETED).

- מיקום: SHARED/data/leagues/league\_2025\_even\_odd/rounds.json

### 9.4.3 נתונים משחק בודד - data/matches/<league\_id>/<match\_id>.json

- מטרה: תיעוד מלא של משחק בודד.

- מערכן: השופט שניהל את המשחק.

- מיקום: SHARED/data/matches/league\_2025\_even\_odd/R1M1.json

קובץ זה הוא ה"תעודת זהות" של המשחק ומכיל:

- מצב המשחק וזמן.

- transcript[] – כל ההודעות שהוחלפו (ההיסטוריה מהלכים).
- .(GAME\_OVER – התוצאה הסופית (תואם ל- result -

#### 9.4.4 היסטוריה שחון – data/players/<player\_id>/history.json

- מטרה: "אזכור אישי" של השחקן.
  - משתמש: השחקן עצמו (לבניית אסטרטגיה).
  - מיקום: SHARED/data/players/P01/history.json
- שחקן חכם יכול להשתמש בקובץ זה כ"אזכור" לשיפור האסטרטגיה שלו:

##### דוגמה: היסטוריה שחון

```
{
  "player_id": "P01",
  "stats": {
    "total_matches": 20,
    "wins": 12, "losses": 5, "draws": 3
  },
  "matches": [
    {
      "match_id": "R1M1",
      "opponent_id": "P02",
      "result": "WIN",
      "my_choice": "even",
      "opponent_choice": "odd"
    }
  ]
}
```

#### 9.5 שכבת הלוגים – logs/

שכבה זו היא "מערכת העצבים" של החברה – היא מאפשרת לנו לראות מה באמת קורה במערכת המבוזרת.

### 9.5.1 לוג ליגה מרכז – logs/league/<league\_id>/league.log.jsonl

- **פורמט:** JSON Lines (כל שורה אובייקט JSON נפרד).
- **משתמשים:** DevOps, תומכה טכנית.
- **מיקום:** SHARED/logs/league/league\_2025\_even\_odd/league.log.jsonl

#### דוגמה: רשומה לוג ליגה

```
{  
    "timestamp": "2025-01-15T10:15:00Z",  
    "component": "league_manager",  
    "event_type": "ROUND_ANNOUNCEMENT_SENT",  
    "level": "INFO",  
    "details": {"round_id": 1, "matches_count": 2}  
}
```

### 9.5.2 לוג סוכן – logs/agents/<agent\_id>.log.jsonl

- **מטרה:** מעקב פר-סוכן לדיבוג.
- **משתמשים:** מפתחי הסוכן.
- **מיקום:** SHARED/logs/agents/P01.log.jsonl

כל סוכן מתעד את ההודעות שהוא שלח ומקבל, מה שמאפשר End-to-End Trace של כל אינטראקציה במערכת.

## 9.6 טבלת סיכום קבצים

### 9.7 שימוש בקבצים המשותפים

כל קובצי הדוגמה המתוארים בסעיפים זה זמינים בתיקייה המשותפת:

L07 / SHARED /

סטודנטים מוזמנים להשתמש בקבצים אלה כבסיס למימוש הסוכנים שלהם. הקבצים כוללים:

- דוגמאות מלאות לכל סוג קובץ.
- נתונים תואמים לפרטוקול league.v2.
- מבנה תיקיות מומלץ לפרויקט.

טבלה 18: סיכום קובצי הקונפיגורציה והנתונים

משתמש	מטרה	נתיב	שכבה
כל הסוכנים	פרמטרים גלובליים	config/system.json	קונפיג
מנהל ליגה	רישום סוכנים	config/agents/	קונפיג
מנהל ליגה	הגדרות ליגה	config/leagues/	קונפיג
שופטים	רישום משחקים	config/games/	קונפיג
סוכנים	ברירות מחדל	config/defaults/	קונפיג
colm	טבלת דירוג	data/.../standings.json	ריצה
מנהל ליגה	היסטוריה מחזוריים	data/.../rounds.json	ריצה
אנליטיקה	פרטי משחק	data/matches/	ריצה
שחקן	היסטוריה אישית	data/players/	ריצה
DevOps	לוג מרכזי	logs/league/	לוגים
פתחים	לוג סוכן	logs/agents/	לוגים

## 9.8 סיכום

ארכיטקטורת שלוש השכבות שהציגו – קונפיגורציה, נתוני ריצה, ולוגים – מספקת את התשתיות הנדרשת לבניית מערכת סוכנים בקנה מידה גדול. כמו בחברה אנושית, גם כאן:

- **הkonfigurציה** היא ה"חוקה" – הכללים היסודיים שכולם מכירים.
- **נתוני הריצה** הם ה"ארכיוון ההיסטורי" – האזכור הקולקטיבי.
- **elogims** הם ה"עתונות" – תיעוד מה שקרה בזמןאמת.

מבנה זה מכין את המערכת לצמיחה של אלפי סוכנים וליגות, תוך שמירה על סדר, עקביות, ויכולת מעקב.