



# DATA AVENUE BEMUTATÓ

Az MTA Cloud felhasználói számára készült  
DataAvenue bemutató v1.0

MTA Cloud csapat  
[info@mta.cloud.hu](mailto:info@mta.cloud.hu)

## Tartalom

Áttekintés.....	2
Lépések.....	2
Occopus telepítése.....	2
Letöltés.....	3
1. Erőforrás szekció kitöltése .....	3
2. Tűzfalszabály létrehozása .....	3
3. Hitelesítés.....	3
4. Klaszter méretének beállítása .....	4
5. Változók szerkesztése.....	4
6. Importálás.....	4
7. Felépítés.....	4
8. A felépült infrastruktúra .....	4
9. Hitelesítő adatok elmentése .....	5
10. IP címek elmentése .....	5
11. Bucket létrehozása.....	5
12. Bucket létrehozásának ellenőrzése.....	6
13. Teszt fájl generálása .....	6
14. Fájl feltöltése .....	6
15. Fájl feltöltésének ellenőrzése .....	6
16. Cél tároló csomópont hitelesítő adatainak elmentése.....	6
17. Fájl másolása.....	7
18. A fájl másolás eredményének ellenőrzése.....	7
19. Bucketben levő fájlok klistázása.....	7
20. Fájl letöltése a bucket-ből .....	7
21. Infrastruktúra törlése: .....	8
Hibák, illetve a projekttel kapcsolatos további igények bejelentése.....	8

## Áttekintés

A DataAvenue egy olyan tároló menedzser szolgáltatás, amely lehetővé teszi a különböző típusú tárolási források (beleértve S3, sftp, GridFTP, iRODS, SRM szerverek) elérését egy egységes felület segítségével. A rendelkezésre álló REST API lehetővé teszi az összes tipikus tárolási műveletet, mint például mappák/bucketek készítése, fájlok/mappák átnevezése vagy törlése, fájlok feltöltése/letöltése, fájlok/mappák másolása vagy áthelyezése a különböző tárolási erőforrások között, akár egyszerűen a "curl" parancs használatával a parancssorból. A DataAvenue automatikusan lefordítja a felhasználók REST parancsát a megfelelő tárolási protokollokra, és kezeli a hosszú távú adatátvitelt a háttérben.

Ebben a bemutatóban létrehozunk egy klasztert, amely két csomóponttípust tartalmaz. A DataAvenue csomóponton a DataAvenue alkalmazás fog futni, és egy előre meghatározott számú tárolási csomóponton (storage node) pedig egy S3 tároló fog futni, hogy képesek legyünk kipróbálni a DataAvenue fájlátviteli szoftvert, mint például bucketek létrehozását, a fájlok letöltését vagy másolását. Ceph és Docker komponenseket használtunk fel a klaszter felépítésére.

### Jellemzők:

- Kétféle csomópont létrehozása kontextualizáció által
- Nova erőforrás használata
- Skálázási paraméterek használata a tároló csomópontok számának méretezéséhez

### Előfeltételek:

- felhő elérése valamilyen Occopus kompatibilis felületen (pl. EC2, Nova, OCCI stb.)
  - Az Occopus eszköz képes az MTA Cloud-on virtuális gépeket indítani, ehhez a nova interfész használatát ajánljuk (ennek címe a felületen megtekinthető: Compute/Access & Security / API Access / Identity )
  - **Fontos: az Occopus eszköz használata jelenleg csak az MTA Cloud SZTAKI ágán működik!**
- a célfelhő tartalmaz egy felhőalapú támogatással rendelkező 16.04 alapú Ubuntu OS képfájlt
  - Ez a képfájl a Compute/Images/Public/Ubuntu 16.04 LTS Cloud image alatt megtalálható

## Lépések

### Occopus telepítése

Az automatikusan felépülő DataAvenue architektúra kiépítése az Occopus eszköz segítségével fog megvalósulni, ezért elsőként telepítenünk kell az Occopus eszközt. Az Occopus orkesztrációs eszközt csupán egyetlen parancs segítségével is telepíteni tudja. A telepítéshez szükséges lépésekről, illetve az Occopus-ról bővebben [az alábbi linken tájékozódhat](#). Javasoljuk, hogy indítson el egy Ubuntu alapú virtuális gépet az MTA Cloud-ban, majd oda telepítse az Occopus eszközt.

## Letöltés

Az Occopus leírók alapján működik. Ezen leírók alapján fogja az Occopus felépíteni a cél felhőben az infrastruktúrát. A DataAvenue telepítéséhez szükséges leírókat elkészítettük a felhasználók számára. Ezeket az alábbi linkről lehet letölteni: [tutorial.examples.dataavenue-cluster](#). Az Occopus-t futtató virtuális gépre telepítjük a leírókat.

**Megjegyzés:** Ebben bemutatóban nova felhő erőforrásokat használunk (az alap bemutató részben található nova oktatóanyagok alapján). Ugyanakkor nyugodtan használhat bármilyen Occopus – kompatibilis felhő erőforrást a csomópontok számára, de javasoljuk, hogy az összes csomópontot azonos felhőben építse fel.

### 1. Erőforrás szekció kitöltése

Nyissa meg a `nodes/node_definitions.yaml` fájlt és szerkessze a csomópontok „`node_def:`” címkével ellátott erőforrás szekcióját:

- Válasszon ki egy [Occopus kompatibilis erőforrás plugin-t](#)
  - Ez az MTA Cloud esetében a bemutatóban szereplő erőforrás plugin-nal azonos, nova erőforrás lesz
- Meghatározhatja [a plugin attribútumainak megfelelő listát](#)
- Követheti a bővítmény [attribútumainak értékeinek összegyűjtésére vonatkozó segítséget](#)
- A plugin-hez tartozó erőforrás sablonokhoz találhat [erőforrás plugin bemutatókat is](#)

Fontos, hogy indítás előtt a `node` definíciós fájlt személyre kell szabnia a felhasználónak. Ebben a fájlban megadjuk azokat az erőforrás azonosítókat, amelyeket használni fogunk, pld.: projekt ID, virtuális gép mérete stb. Ezeket az azonosítókat nem tudjuk megadni a felhasználó helyett, azonban azokat az MTA Cloud felületéről könnyedén össze lehet szedni. Ehhez részletes segítség [ezen linken](#), vagy az [alábbi dokumentációban](#) található. A példában található letölthető csomag a Nova plugin erőforrás sablont tartalmazza (ezt használjuk az MTA Cloud használatához is).

### 2. Tűzfalszabály létrehozása

Az infrastruktúra komponensei egymáshoz kapcsolódnak, ezért több port tartományt is ki kell nyitni a virtuális gépek számára. Jelentkezzünk be az [MTA Cloud OpenStack felületére](#). A „Compute/Access&Security” menüpont alatt létrehozhatunk új tűzfalszabályt a „Create Security Group” gombra kattintva. Létrehozás után szerkeszteni tudjuk a tűzfalszabályt a „Manage Rules/Add Rule” gombra kattintva. Az alábbi portokat adjuk hozzá a tűzfalszabályhoz:

- TCP 22 (SSH)
- TCP 80 (HTTP)
- TCP 8080

### 3. Hitelesítés

Az Occopus számára szükség van az MTA Cloudban használt felhasználónév/jelszó párosra, hogy az hitelesíteni tudja magát és képes legyen virtuális gépeket/infrastruktúrákat létrehozni egy adott projekt alatt. Kérjük, ellenőrizze, hogy az Occopus telepítése során helyesen adta-e meg a hitelesítési adatait! Az autentikációs adatok megadásához [az alábbi linken](#) talál segítséget.

## 4. Klaszter méretének beállítása

Ha szükséges, frissítse a tároló csomópontok számát (storage node). Ehhez szerkessze az **infra-dataavenue.yaml** fájlt és módosítsa a „min” paramétert a „scaling” kulcsszó alatt. A skálázás az az intervallum, amelyben a csomópontok száma megváltozhat (min, max). Jelenleg a minimális érték 2-re van állítva (ami az indításkor a kezdeti szám lesz).

```
- &S
  name: storage
  type: storage_node
  scaling:
    min: 2
```

**Fontos:** Ne feledje, hogy az Occopusnak legalább egy csomópontot el kell indítani minden egyes csomóponttípusból, hogy megfelelően működjön az infrastruktúra, valamint a méretezés csak a példában található tároló csomópontokra (storage\_node) alkalmazható!

## 5. Változók szerkesztése

Ha szükséges akkor szerkessze a „variables” szekciót az **infra-dataavenue.yaml** fájlban. Az alábbi értékek paraméteres beállítására van lehetőség:

- **storage\_user\_name** az S3 tárolón létrehozott felhasználó neve
- **access\_key** az S3 tárolón létrehozott felhasználóhoz tartozó access key
- **secret\_key** az S3 tárolón létrehozott felhasználóhoz tartozó secret key

## 6. Importálás

Importáljuk a személyre szabott leírókat az Occopus adatbázisába:

```
occopus-import nodes/node_definitions.yaml
```

Győződjön meg róla, hogy a megfelelő virtualenv aktiválva van! Amennyiben ezt még nem tette volna meg korábban, az alábbi parancs segítségével aktiválható az Occopus virtuális környezete:

```
source occopus/bin/activate
```

**Fontos:** Az Occopus akkor veszi fel a csomópont definíciókat az adatbázisból, amikor felépíti az infrastruktúrát, így mindig importálásra van szükség, ha a node definíciós fájl, vagy bármelyik fájl (pld.: kontextualizációs fájl) megváltozik!

## 7. Felépítés

Indítsa el az infrastruktúra felépítését az alábbi paranccsal:

```
occopus-build infra-dataavenue.yaml
```

## 8. A felépült infrastruktúra

Sikeres lefutás után a virtuális gépek IP címei, csomópont azonosítói, valamint az infrastruktúra azonosítója megjelenik a log üzenetek alján, listába szedve. Az infrastruktúra azonosító elmenthető, vagy lekérdezhető az occopus-maintain parancs segítségével:

```
List of nodes/ip addresses:
dataavenue:
  <ip-address> (dfa5f4f5-7d69-432e-87f9-a37cd6376f7a)
storage:
  <ip-address> (cae40ed8-c4f3-49cd-bc73-92a8c027ff2c)
  <ip-address> (8e255594-5d9a-4106-920c-62591aab899)
77cb026b-2f81-46a5-87c5-2adf13e1b2d3
```

## 9. Hitelesítő adatok elmentése

Az S3 tároló csomópontokon egy előre definiált paraméterekkel rendelkező felhasználó jön létre. Az **access\_key** lesz a felhasználónév és a **secret\_key** lesz a jelszó, amelyek az **infra-dataavenue.yaml** fájlban vannak előre meghatározva. A felhasználói hitelesítő adatok „**credentials**” nevű fájlba történő mentéséhez használja az alábbi parancsot:

```
echo -e 'X-Key: 1a7e159a-ffd8-49c8-8b40-549870c70e73\nX-Username:
A8Q2WPCWAELW61RWDGO8\nX-Password:
FWd1mccBfnw6VHa2vod98NEQktRCYlCronxb01aQ' > credentials
```

**Megjegyzés:** Ez a lépés hasznos lesz a későbbi a DataAvenue használatához szükséges curl parancsok lerövidítéséhez.

## 10. IP címek elmentése

Mentse el a virtuális gépek IP címét változóba, hogy a későbbi parancsok egyszerűbbek legyenek.

```
export SOURCE_NODE_IP=[storage_a_ip]
export TARGET_NODE_IP=[storage_b_ip]
export DATAAVENUE_NODE_IP=[dataavenue_ip]
```

## 11. Bucket létrehozása

Hozzon létre egy bucket-et mindegyik S3 tároló csomóponton:

```
curl -H "$(cat credentials)" -X POST -H "X-URI:
s3://$SOURCE_NODE_IP:80/sourcebucket/"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/directory
```

```
curl -H "$(cat credentials)" -X POST -H "X-URI:
s3://$TARGET_NODE_IP:80/targetbucket/"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/directory
```

**Megjegyzés:** A bucket-ek nevének legalább három betű hosszúságúnak kell lennie. Itt, a példában a bucket neve a forrás S3 tároló csomóponton a „**sourcebucket**” lesz, a bucket neve a cél S3 tároló csomóponton pedig „**targetbucket**” lesz.

## 12. Bucket létrehozásának ellenőrzése

Ellenőrizze a bucket létrehozást azáltal, hogy kilistázza a bucketeket mindegyik tároló csomóponton:

```
curl -H "$(cat credentials)" -H "X-URI: s3://$SOURCE_NODE_IP:80/"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/directory
```

A helyes eredmény: ["sourcebucket/"]

```
curl -H "$(cat credentials)" -H "X-URI: s3://$TARGET_NODE_IP:80"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/directory
```

A helyes eredmény: ["targetbucket/"]

## 13. Teszt fájl generálása

A DataAvenue fájl átviteli szoftver teszteléséhez szükség lesz egy fájl létrehozásához, amelyen kipróbálhatjuk a fájl átviteli műveleteket. Az alábbi parancs segítségével egy előre meghatározott méretű fájlt tud generálni, most ennek a fájlnak a mérete 1 megabájt lesz:

```
dd if=/dev/urandom of=1MB.dat bs=1M count=1
```

## 14. Fájl feltöltése

Töltse fel a generált „1MB.dat” nevű fájlt a forrás tároló csomópontra:

```
curl -H "$(cat credentials)" -X POST -H "X-URI:
s3://$SOURCE_NODE_IP:80/sourcebucket/1MB.dat" -H 'Content-Type:
application/octet-stream' --data-binary @1MB.dat
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/file
```

## 15. Fájl feltöltésének ellenőrzése

Ellenőrizze a fájl feltöltést a forrás tároló csomóponton található „sourcebucket” nevű bucket kilistázásával:

```
curl -H "$(cat credentials)" -H "X-URI:
s3://$SOURCE_NODE_IP:80/sourcebucket"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/directory
```

A helyes eredmény: ["1MB.dat"]

## 16. Cél tároló csomópont hitelesítő adatainak elmentése

Mentse el a cél tároló csomópont hitelesítő adatait a „target.json” nevű fájlba, hogy a későbbi parancsok lerövidüljenek:

```
echo
"{target:'s3://"$TARGET_NODE_IP":80/targetbucket/', overwrite:true, creden
tials:{Type:UserPass, UserID:"A8Q2WPCWAELW61RWDG08",
UserPass:"FwD1mccBfnw6VHa2vod98NEQktRCY1Cronxb01aQ"}}" > target.json
```

## 17.Fájl másolása

Másolja át a feltöltött „1MB.dat” nevű fájlt a forrás tároló csomópontról a cél tároló csomópontra:

```
curl -H "$(cat credentials)" -X POST -H "X-URI:
s3://$SOURCE_NODE_IP:80/sourcebucket/1MB.dat" -H "Content-type:
application/json" --data "$(cat target.json)"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/transfers > transferid
```

A helyes eredmény: [transfer\_id]

## 18.A fájl másolás eredményének ellenőrzése

Ellenőrizze a fájl másolás eredményét a **transfer\_id** lekérdezésével, amelyet a korábbi, másolás parancs kiadásával kaptunk:

```
curl -H "$(cat credentials)"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/transfers/$(cat
transferid)
```

A következő eredmény egy sikeres másolást jelent a forrás csomópontról a cél csomópontra (lásd: status:DONE, azaz eredmény: KÉSZ)

```
"bytesTransferred":1048576,"source":"s3://[storage_a_ip]:80/sourcebucket
/1MB.dat","status":"DONE","serverTime":1507637326644,"target":"s3://[sto
rage_b_ip]:80/targetbucket/1MB.dat","ended":1507637273245,"started":1507
637271709,"size":1048576
```

## 19.Bucketben levő fájlok listázása

Listázza ki a cél csomóponton található bucket-et, hogy ellenőrizzük, a másolás tényleg sikeres volt:

```
curl -H "$(cat credentials)" -H "X-URI:
s3://$TARGET_NODE_IP:80/targetbucket"
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/directory
```

A helyes eredmény: ["1MB.dat"].

## 20.Fájl letöltése a bucket-ből

Töltse le az átmásolt fájlt a cél csomópontról:

```
curl -H "$(cat credentials)" -H "X-URI:
s3://$TARGET_NODE_IP:80/targetbucket/1MB.dat" -o download.dat
http://$DATAAVENUE_NODE_IP:8080/dataavenue/rest/file
```



## 21. Infrastruktúra törlése:

Végül, lebonthatja az infrastruktúrát az **occopus-build** által visszaadott infrastruktúraazonosító használatával:

```
occopus-destroy -i 77cb026b-2f81-46a5-87c5-2adf13e1b2d3
```

**Megjegyzés:** Ebben a bemutatóban csak HTTP protokollt használtunk. A DataAvenue azonban képes támogatni a HTTPS protokollt is a 8443-as porton, így a tárolók elérhetőek egy biztonságos HTTP protokollon keresztül is, ha telepítünk pld. HAPROXY-t a csomópontokra.

### [Hibák, illetve a projekttel kapcsolatos további igények bejelentése](#)

Az MTA Cloud szolgáltatással kapcsolatos kommunikáció és támogatás e-mail formájában történik. A közös e-mail cím: [info@cloud.mta.hu](mailto:info@cloud.mta.hu). Az erre az e-mail címre bejelentett hibákból, igényekből egy bejelentési űrlap generálódik, melyet az MTA Cloud csapat kijelölt tagja kezel.