Digital Sky Platform

Technical Architecture (v1.0, Draft)

Table Of Contents

Definition of Terms

CAR	Civil Aviation Requirements
DGCA	Directorate General of Civil Aviation
MHA	Ministry of Home Affairs
RPAS	Remotely Piloted Aircraft System(s)
UA	Unmanned Aircraft
UAPL	Unmanned Aerial Pilot License
UAS	Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
UIN	Unique Identification Number
UTM	UAS Traffic Management
ASP	Application Service Provider
DSP	Digital Sky Service Provider

Introduction

Imagine a future where drones augment human capabilities. Drones could help farmers prioritize where to apply fertilizer. They also could help energy companies monitor their infrastructure. Drones could even enable emergency response teams to quickly map the extent of damage after natural disasters. In the future, drones could deliver packages, streamline agriculture management, reinvent human mobility, and even save lives. Therefore, it's critical that we have in place a seamless and secure technology and regulatory framework to integrate this new technology into the Indian airspace.

Currently, all civil drone operations are banned awaiting release of regulations - this is stifling the UAV & related industries. Paper based regulations are a time-taking, uncertain and unwieldy process for both regulator and end-user. Moreover, they offer very few tools to the regulator to enforce rules. Knowing this, Unmanned Aerial Operators skip getting the requisite training for their pilots, as well as skip the permissions and fly their UAVs with impunity. Further, the UAVs pose a risk to security and safety such as flying close to airports, military bases, monuments of national importance, etc.

Jayant Sinha, Honorable minister of state for civil aviation, said "The idea is to allow unfettered and unrestricted use of these drones so that we can develop new and unique applications." In light of the same, we need to create a "future ready" regulatory and technology framework that catalyses a mass flourishing of innovations. The key to widespread adoption of drones will be the ease of process in getting the paperwork around licenses and certification done. If not, the UAV industry will stay stunted. Therefore, it's critical to convert the paper process to digital, thus taking a paperless and presenceless approach to drone permission.

India needs a world class drone policy backed by a world class technology platform enabling seamless execution of the policy for all the stakeholders. We need to implement from the ground up a modern, automated, fully digital digital sky platform. On this technology backbone, drone operators can apply for permissions in almost real-time from their mobile phones, and an automated process will grant permission, in the form of a digitally signed certificate, if they are clear of all designated no-UAV zones.

A proactive approach to enforcement of safety and security guidelines is done by ensuring manufacturer does not allow take-off without signed digital certificate from Digital Sky and logs all flight plans with the Digital Sky. Further, incident reporting & analytics tools will be available to the Digital Sky to monitor for potential hazards.

We envision a future, when millions of UAVs are flying across the country, without significantly increasing the regulatory burden. The Digital Sky can be extended in the future to carry out autonomous flights, automated UAV traffic control, air taxis, besides other use-cases.

Mission

The mission of this framework is to create a completely digital, paperless, and presenceless process, thus fast-forwarding to a future of on-demand seamless permissions for drones, operators, and pilots.

Vision

The vision is to create a digital infrastructure that will support safe, efficient, and secure access to Indian airspace for millions of drones.

Note: The purpose of this document is to clearly state the strategy for the Technology and API Details. It also provides detailed examples, use cases, and flows.

The regulatory and procedural aspects pertaining to UAVs are out of scope for this document.

This is a draft specification and it may undergo changes before becoming final specifications based on feedback from the ecosystem.

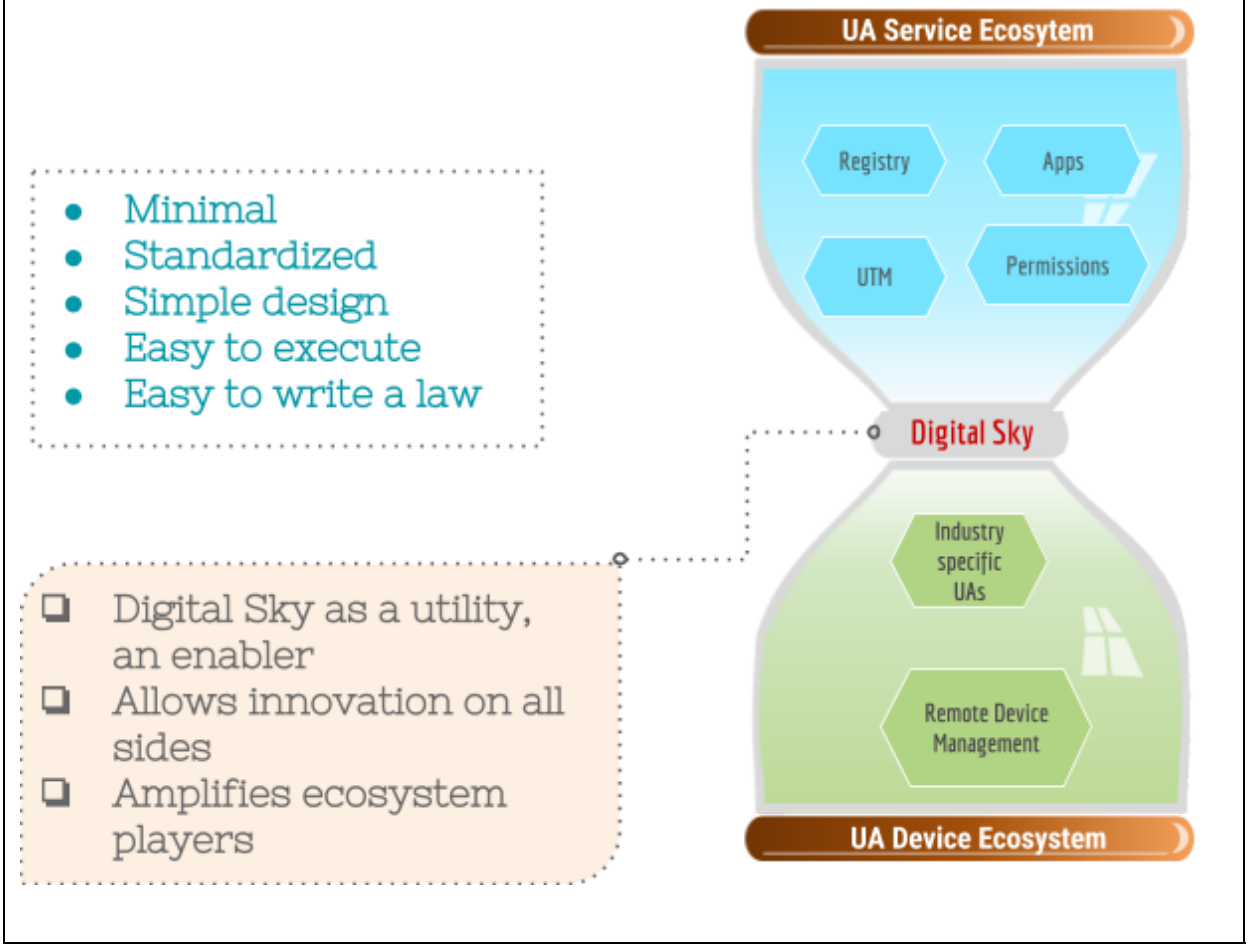
Guiding Design Principles

- **Universal Identity**: The technical framework should leverage universal, authenticable, non-repudiable, and digital identities to allow interoperability across all actors (pilot, manufacturer, operator, drone, etc) in the system.
- **User-Centric:** The framework should be designed by placing the user in the centre, thus only adopting approaches that are convenient and easy for doing business.
- **Granular Control and Digital Enforceability**: The framework should allow users to set permissions and rights for permission access at a granular level and the same must be enforced digitally, thus generating verifiable audit trails.
- **Open Platform and Open Standards Based:** The framework should use open technology and legal standards available in the country. It should be agnostic to applications, programming languages, and platforms.
    - People should have programmatic interfaces for sharing and accessing the information available to them. The specifications for these interfaces should be published and made available and accessible to everyone.
    - Use of open standards to ensure interoperability
    - Allow the ecosystem to build on top of standard APIs
    - Vendor neutral, using of commodity computing and open source
    - Designed to work with any device, any form factor, any network
- **Security by Design**
    - The software and systems must be designed from the ground up to be secure. End-to-End security of data (PKI, DSC, tamper detection).
- **Privacy by Design**
    - Proactive not reactive; Preventative not remedial
    - Privacy as the default setting
    - Privacy embedded into design
    - Full functionality – positive-sum, not zero-sum
    - End-to-end security – full lifecycle protection
    - Visibility and transparency – keep it open
    - Respect for user privacy – keep it user-centric
    - Minimal Data, APIs, and Data Anonymization
- **Design for Scale**
    - High Performance and High Availability
    - Every component needs to scale to large volumes
    - 100's of millions of transactions and billions of records
    - Fully multi location distributed architecture for horizontal scale
- **Ecosystem Driven Approach:** An ecosystem approach is necessitated such that the interfaces between the partners and systems are well defined and standardized. Hence, there must exist a technology backbone that would hold together this partner ecosystem.

- **Trustable and IT Act Compliant**: Use digital signatures to guarantee integrity of access permissions given by users in permission flows. This avoids security issues faced by existing approaches and also makes the framework fully legal under the IT Act.
- **Minimalist and Evolutionary Design**: The design should be simple and minimalistic. It should not present adoption barriers for the ecosystem. The design of the systems should be evolutionarily - their capabilities should be built incrementally while allowing for rapid adoption.

Conceptual Architecture

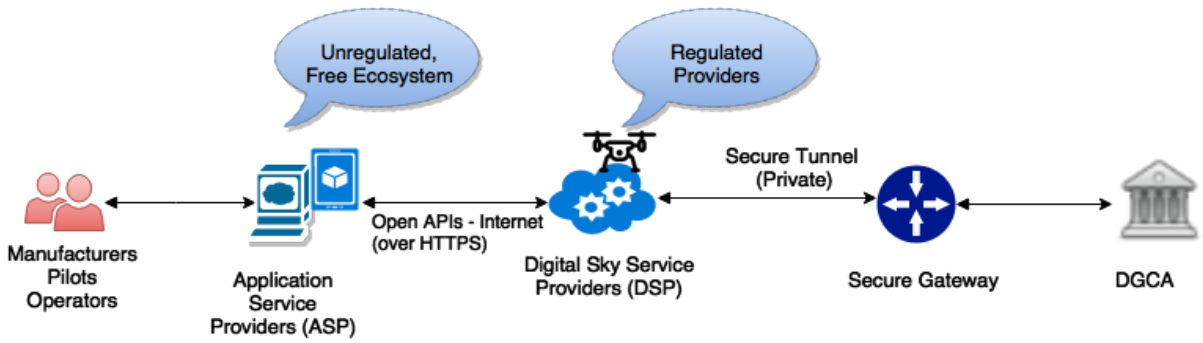
Digital Sky Hourglass Architecture



The key driver for Digital Sky is to foster innovation across the UA ecosystem - both on the devices side and on the services/application side. The key is to allow a diverse set of

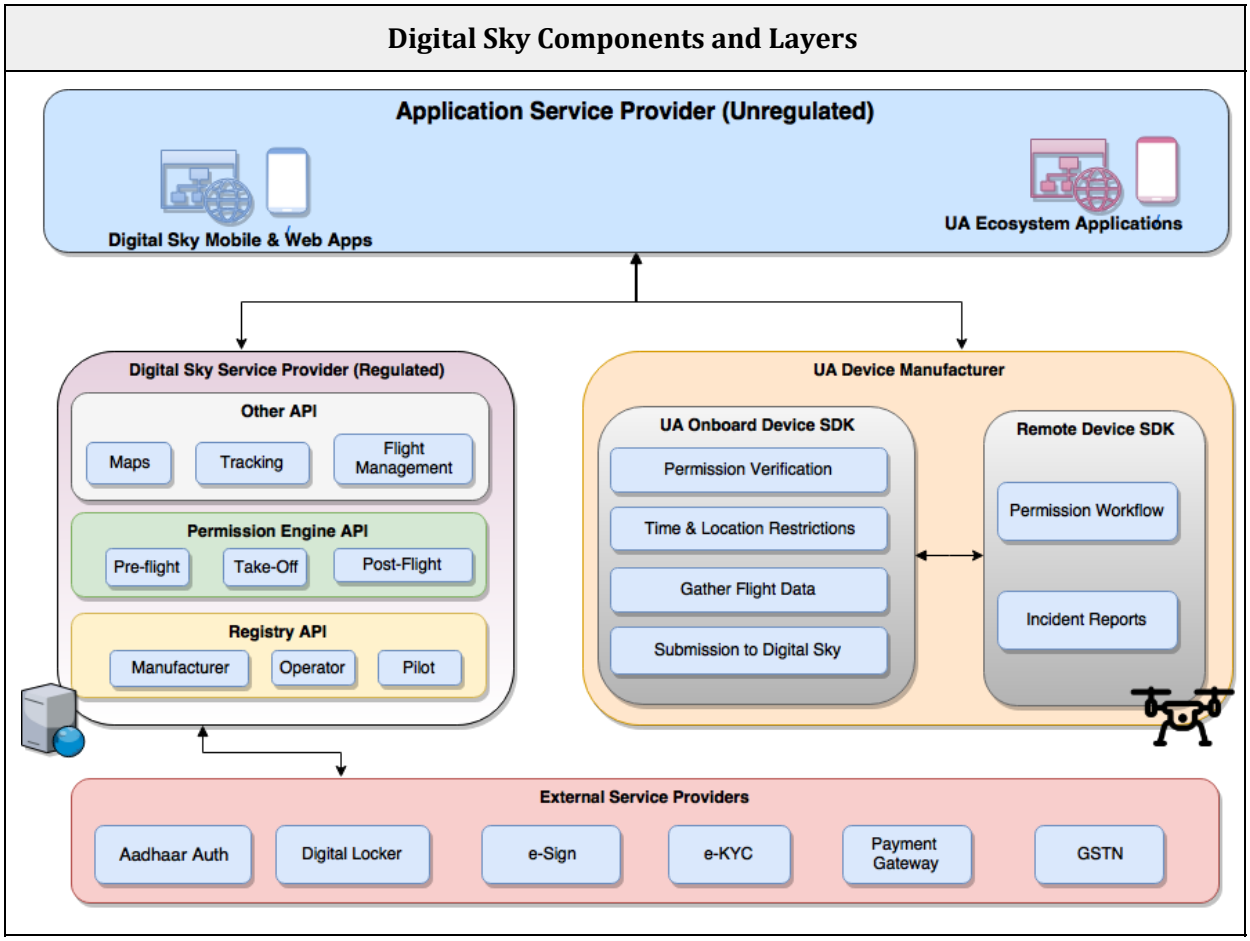
service providers and application providers that can cater to specific industries or use cases. They are brought together in a minimalistic system design involving simple Open APIs. This way Digital Sky is more of a public utility, an enabler.

Digital Sky Ecosystem Architecture



To achieve rapid adoption and innovation in the UA ecosystem, a federated model of service access is recommended:

- The core services of the Digital Sky system - Permission, Registry etc. should be provided through Digital Sky Service Providers (DSP). These could be government or private organizations that implement the core API and link to DGCA through a secure gateway. In other words, the core system is not directly exposed to the internet. The registration of DSPs is tightly managed by DGCA, which should have specific selection criteria, infrastructure and security policies. DSPs could serve specific industry or use cases.
- Application Service Providers (ASP) develop web or mobile applications that are leveraged by operators, pilots and other users of the UA ecosystem. They partner with relevant DSPs to provide value-added services to their users over the internet. ASPs typically are startups or enterprises, but are not regulated in any way. Connectivity between the mobile app and DSP service should be over a secure transport layer like HTTPS.



The core pieces of the Digital Sky architecture represented by the diagram above are:

1) **Digital Sky Service Provider API**: At the heart of the system is a set of Open APIs that service providers can implement covering the following areas:
    a) Permissions Engine: Deals with all permissions related capability. Allows to submit a permission request, provides an approved permission artifact, revokes permission, does pre-flight, post-flight and take-off time permission evaluation.
    b) Registry: Provides a registry for manufacturers, operators and pilots. Provides a seamless way to register, transfer revoke permits.
    c) Other: Provides capabilities for mapping (geo-fencing, etc), capabilities to manage flights (add, delete, update flight information) and UA tracking.
2) **UA Manufacturer SDK and specifications**
    a) UA onboard device library: These are a set of capabilities that all UA manufactures should embed in the UA software. This includes permission verification, enforcing

time and location restriction, collecting and submitting flight information to the Remote device library

b) Remote device library: This is where all of the intelligence and connectivity lies. While the UA itself will not communicate directly with Digital Sky APIs, the remote device library will support the permission, flight logging and Incident related workflows. All communication of data and enforcement of the permissions will be channeled through the remote device library.

3) **Application Service Providers**: All access to the Digital Sky system is only facilitated through a set of Open APIs. Mobile Apps and Websites would be the interface for interacting with Digital Sky for all workflows. ASPs partner with the relevant DSPs to provide value-added services in this ecosystem.

4) External Service Providers: These are providers for key service requirements like authentication, e-KYC, e-Sign and Digital Locker capabilities (India Stack). Integration with a payment gateway is required for collecting fees in a cashless way..

Principles behind the Approach

- **No Permission, No Take-off**
- **Paperless, Cashless, and Presenceless**: The architecture should natively support a Paperless, Cashless and a Presence-less transaction model

Ecosystem

The main entities in the Digital Sky Ecosystem and their responsibilities are:

Entity	Responsibilities
DGCA	1. Owns the Registry of manufacturers, operators, UAs and pilots 2. Owns the specifications for APIs and UA Onboard & remote device SDK 3. Tests UAs against specifications and certifies them. 4. Approves or Rejects permission requests from UA operators
DSP	1. Hosts the Digital Sky services based on the Open API specification. 2. Bridge between DGCA and the user ecosystem
ASP	1. Develops the Mobile Apps and Website that can be used by

	operators, pilots and manufacturers.
UA Manufacturer	1. Manufactures UAs with software compliant with the Digital Sky Onboard device SDK 2. Registers the UINs with DGCA
UA Operator	1. Registers for operating UA with Digital Sky 2. Applies for flight permissions
UA Pilot	1. Applies for UAPL 2. Submits flight logs and incident reports

Functional Flows

Core Workflow Requirements

Manufacturers

Manufacturers only sell UAs which are approved by DGCA:

1. Onboard software is compliant with the proposed SDK specifications
2. All DGCA tests pass
3. The UA UINs have to be registered with DGCA

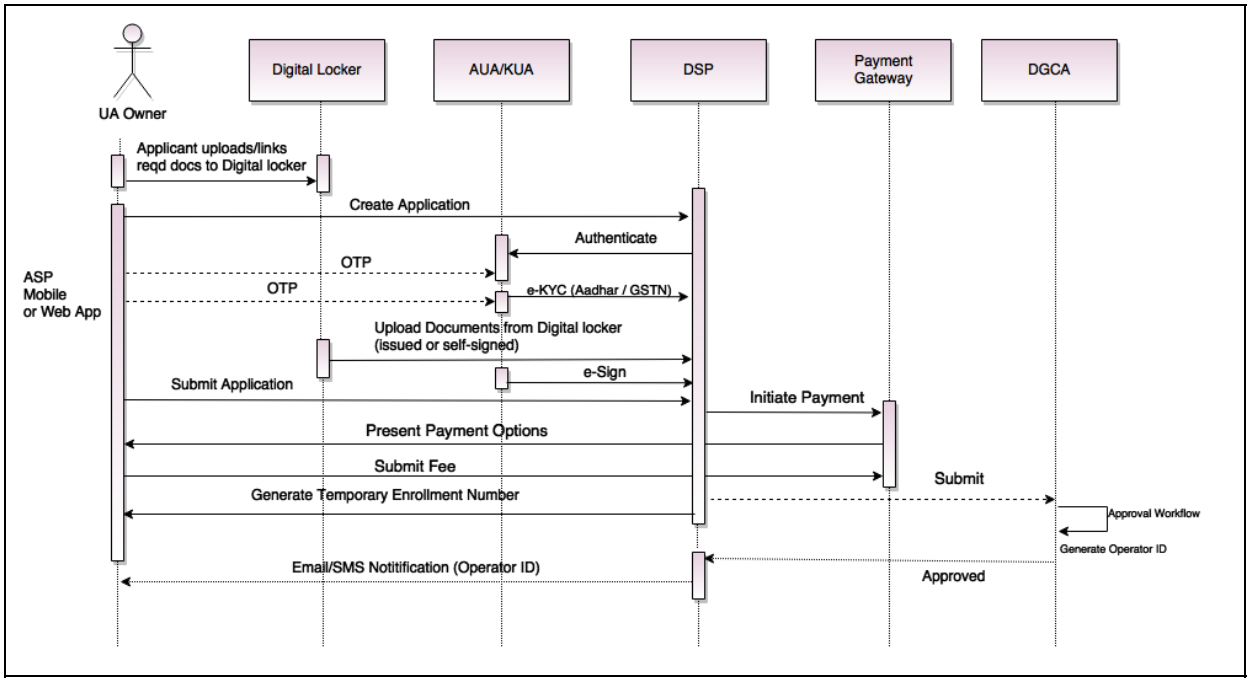
Operators

1. All operators should register before operating any UA
2. e-KYC process has to be completed
3. Should possess an approved permission artifact before any flight can be made

Pilots

1. Should have a valid UAPL before operating any UA
2. Should authenticate (using two-factor authentication) before any flight
3. Should digitally sign and submit flight logs and incident reports to DGCA

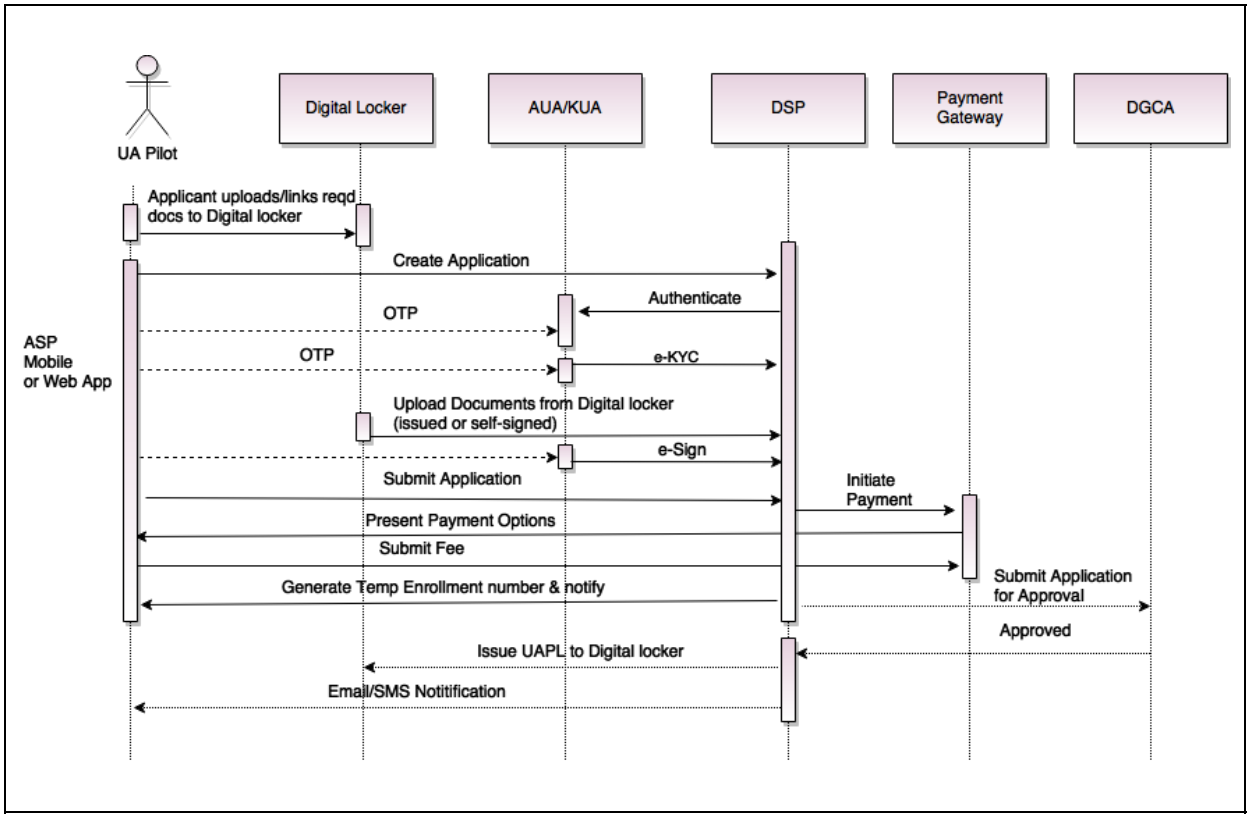
Unmanned Aerial Operator Registration



1. Operator registration may require several documents to be in order. For documents that cannot be issued to a Digital Locker, the applicant should be able to upload a copy of these documents to his account and signs them digitally.
2. The applicant uses the ASP mobile app or a website to login and submit the registration. The website and app should support a "Upload from Digital locker" option so that digitally signed documents or issued documents can be uploaded directly.
3. An e-KYC is required for the operator. Aadhar e-KYC could be used for individuals and GSTN for businesses.
4. Once the documents are uploaded, the application should be digitally signed by the applicant (like Aadhaar e-Sign) and submitted to the DSP.
5. The request is then redirected to a payment gateway through which required fee is collected from the user.

6. Upon successful payment, the DSP sends the application to DGCA, which has an approval workflow. The DSP generates a temporary enrollment number.
7. Once the application is approved, a unique Operator ID is generated and the operator is notified.

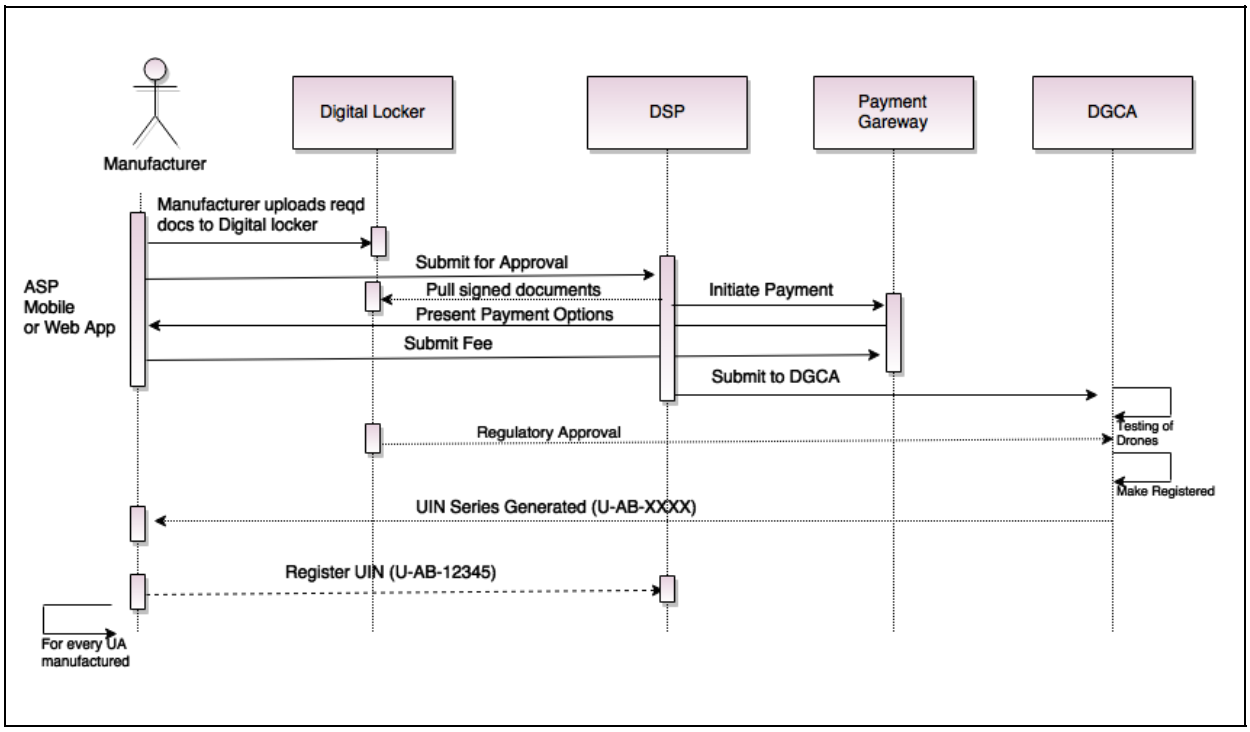
Unmanned Aerial Pilot Licence (UAPL)



1. The applicant uploads a copy of documents to her account and signs them digitally (for documents that cannot be issued to a Digital Locker by a department). The documents that may be required for this process are:
    a. Training Certificates
2. The applicant uses the ASP mobile app or a website to login and submit the requisition for UAPL. The website and app should support a "Upload from Digital locker" option so that digitally signed documents or issued documents can be uploaded directly.
3. For establishing proof of address and proof of identity, an e-KYC process (like Aadhar eKYC) can be leveraged.

4. Once the documents are uploaded, the application should be digitally signed by the applicant (like Aadhaar e-Sign) and submitted to the DSP.
5. The request is then redirected to a payment gateway through which required fee is collected from the user.
6. Upon successful payment, the DSP submits the application for DGCA approval and generates a temporary enrollment number.
7. Subsequently, post approval (or rejection) from DGCA, the applicant is notified. The digitally signed UAPL is issued to the Digital Locker

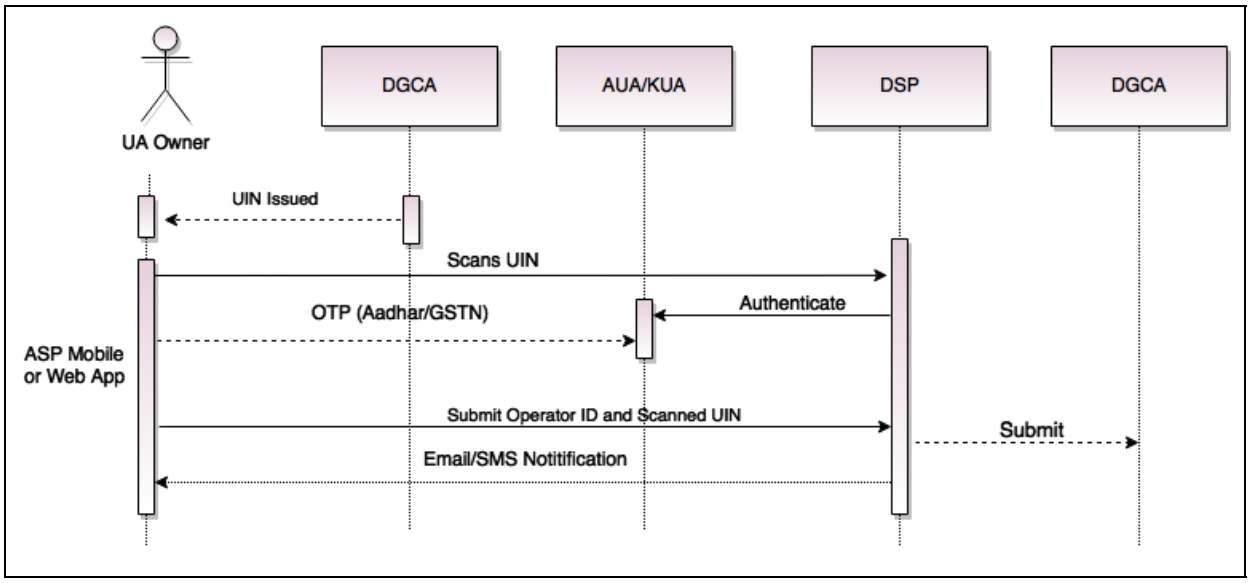
Unmanned Aircraft and Make Registration (UIN Series)



1. The manufacturer of a UA uploads a copy of documents to its account and signs them digitally (for documents that cannot be issued to a Digital Locker by a department). The documents required for this process are:
    a. Operating Manual
    b. Technical Specifications
2. The applicant uses the ASP mobile app or a website to login and submit the registration for UIN series.

3. The request is then redirected to a payment gateway through which required fee is collected from the user.
4. Upon successful payment, the application is submitted to the DGCA which conducts several tests, as outlined in the Test Structure for UAS Certification, on the UA and after all checks pass and with all regulatory approvals/clearances, creates a registration with a new UIN series. This series will be employed for all UAs this particular manufacturer.
5. For every UA manufactured, the manufacturer submits a request to Digital Sky for registering the UIN of the UA. Subsequent permission would be provided only to those UAs which are registered by the manufacturer. This would ensure that unauthorized, illegal UAs don't get permission to take off.

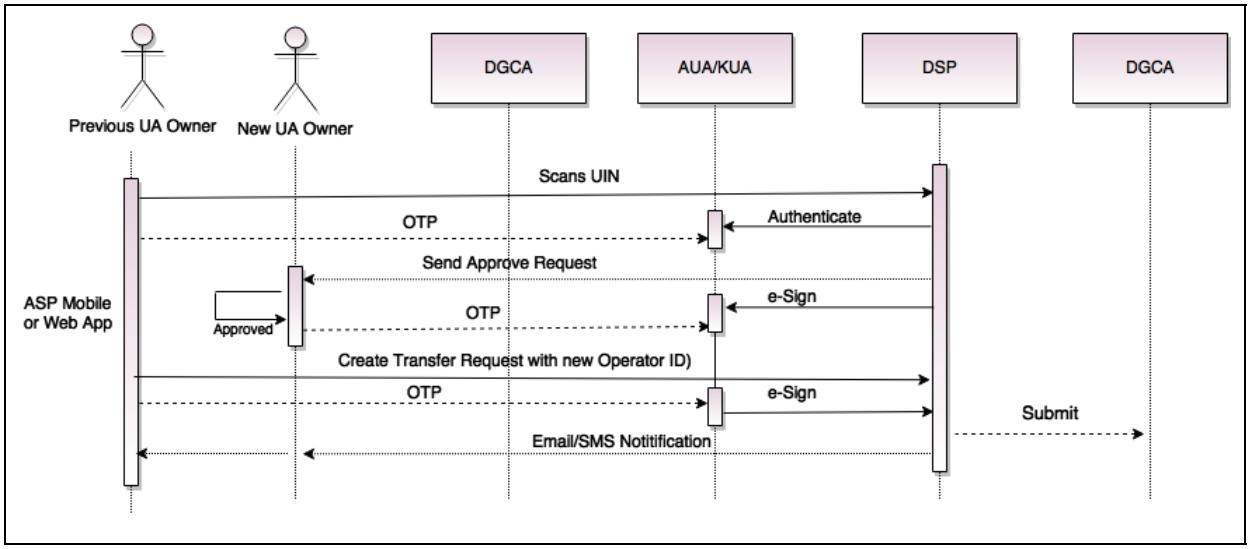
UIN Linking with Operator



Linking of the UIN and Operator ID is important from the perspective of permission request process and would also be required during sale or transfer of the UA. The basic steps are:

1) The owner scans the UIN to start the linking process.
2) Then the owner authenticates (Aadhaar OTP can be used for individuals, GSTN API for businesses) to complete the linking process
3) The owner then gets an SMS or email notification.

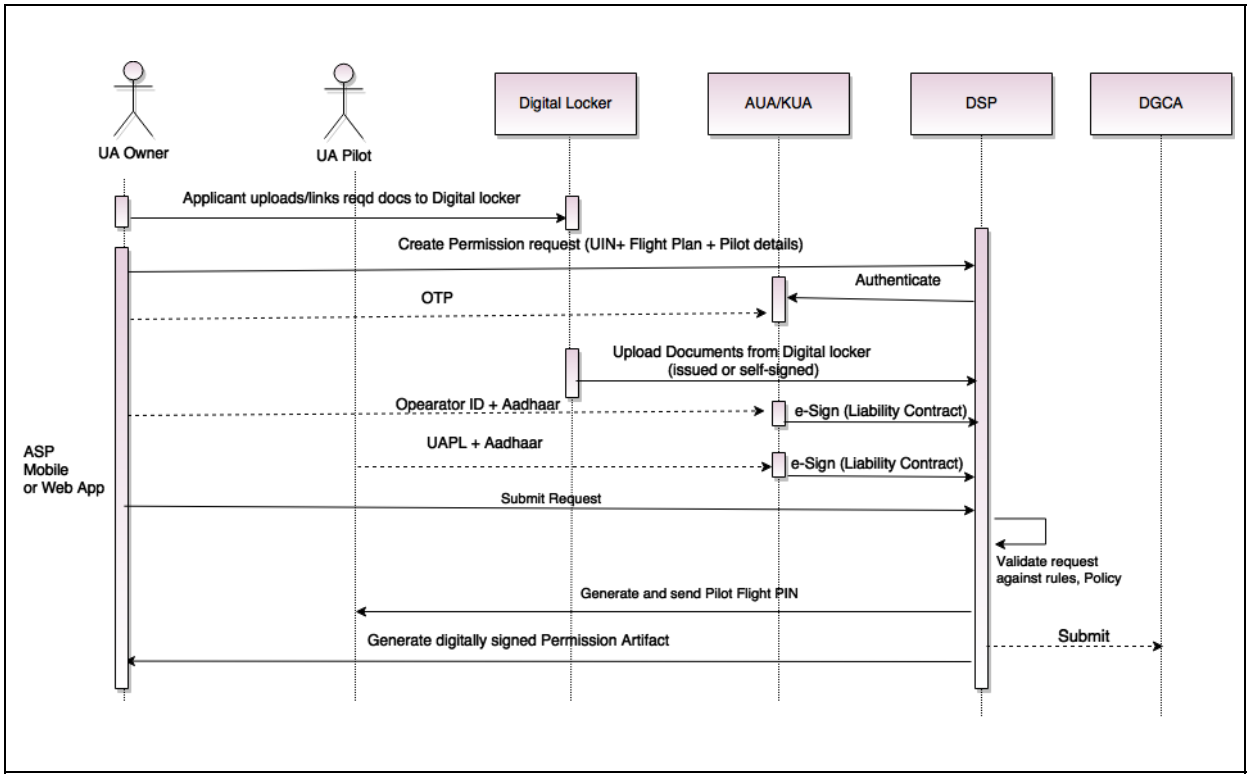
Sale or Transfer of UA



In case of a sale or transfer of UA, the basic steps that has to be followed are:

1) The previous owner scans the UIN, authenticates with Aadhaar and creates a transfer request providing the Operator ID of the transferee.
2) The new owner (transferee) gets a notification for the transfer request.
3) The new owner can approve the request, and e-Sign the transfer agreement (digital signature like Aadhaar e-sign). If approved, the transferor can proceed with the next steps.
4) The transferor then authorizes the transfer by the usage of his digital signature (like Aadhaar e-Sign).
5) Both parties get an SMS or email notification.

Automated Permission Issuance



1) The UA owner uploads a copy of required documents to his account and signs them digitally (for documents that cannot be issued to a Digital Locker by a department).
2) The UA owner then creates a request in the Digital Sky app or website and furnishes all details about the flight, including but not limited to:
    a) Flight Details:
        i) GPS coordinates
        ii) Frequencies used
        iii) Start Time and end times of flight
        iv) Number of flight passes
        v) Purpose of the flight
        vi) Payload details, weight etc
    b) UA, Owner and Pilot details
        i) UIN of the UA
        ii) Operator ID of the owner
        iii) UAPL of the pilot
        iv) Address and contact details of the owner (KYC details)

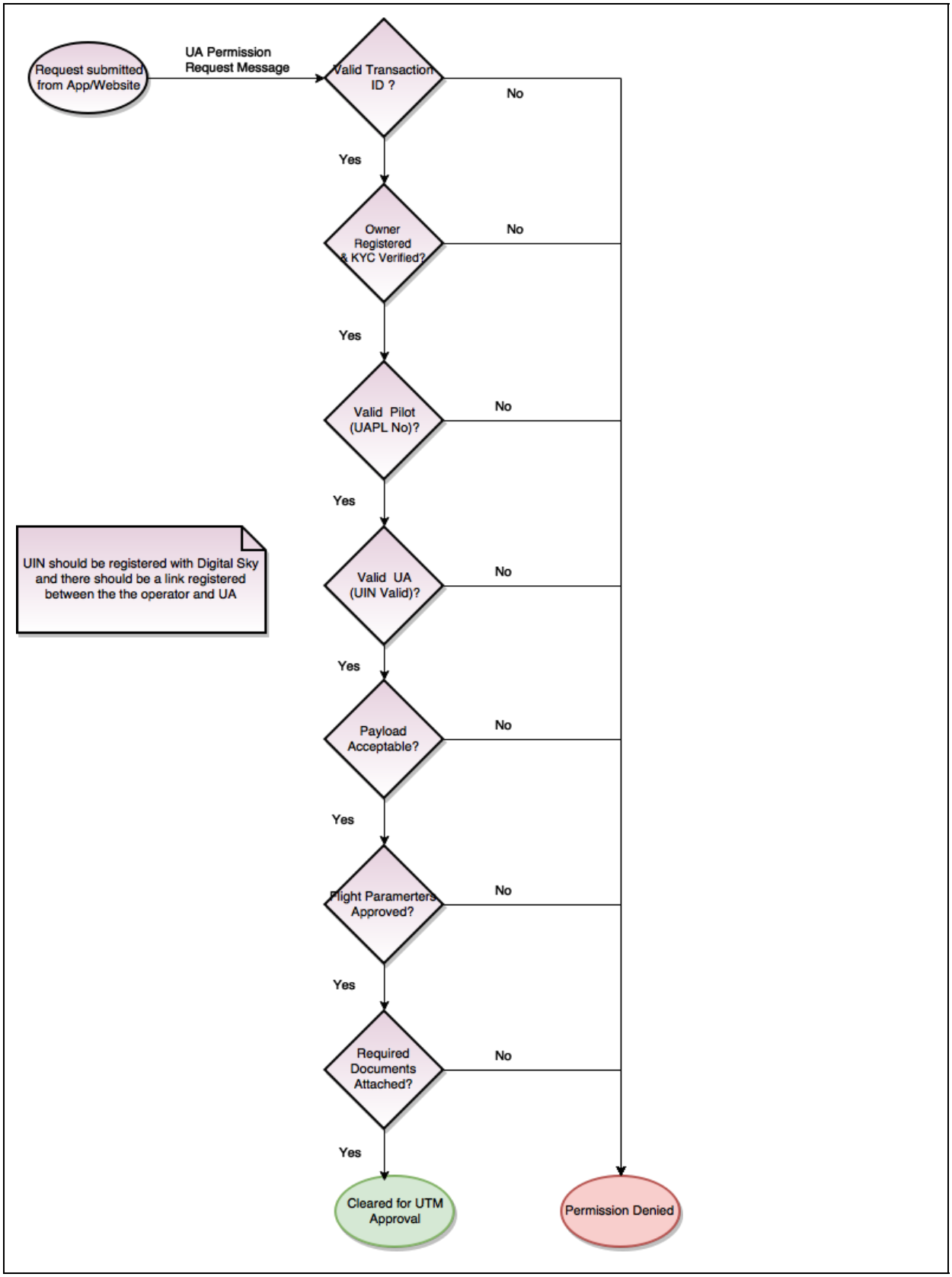
3) The UA owner would be required to sign the application with a liability contract. This can be done digitally using Aadhaar e-sign using Aadhaar number and Operator ID
4) The pilot would need to sign the liability contract digitally using Aadhaar and UAPL
5) The required documents (self-signed) or issued is then linked to the request from Digital Locker and submitted to Digital Sky
6) Once submitted to Digital sky, the request goes through some checks and approval process. This is detailed in the next section.
7) On approval, a digitally signed permission artifact is generated..
8) Once approved, Digital sky also generates a Pilot Flight PIN. This would be baked into the permission artifact. The pilot will be required to key in the PIN before the flight take off.

Approval Workflow

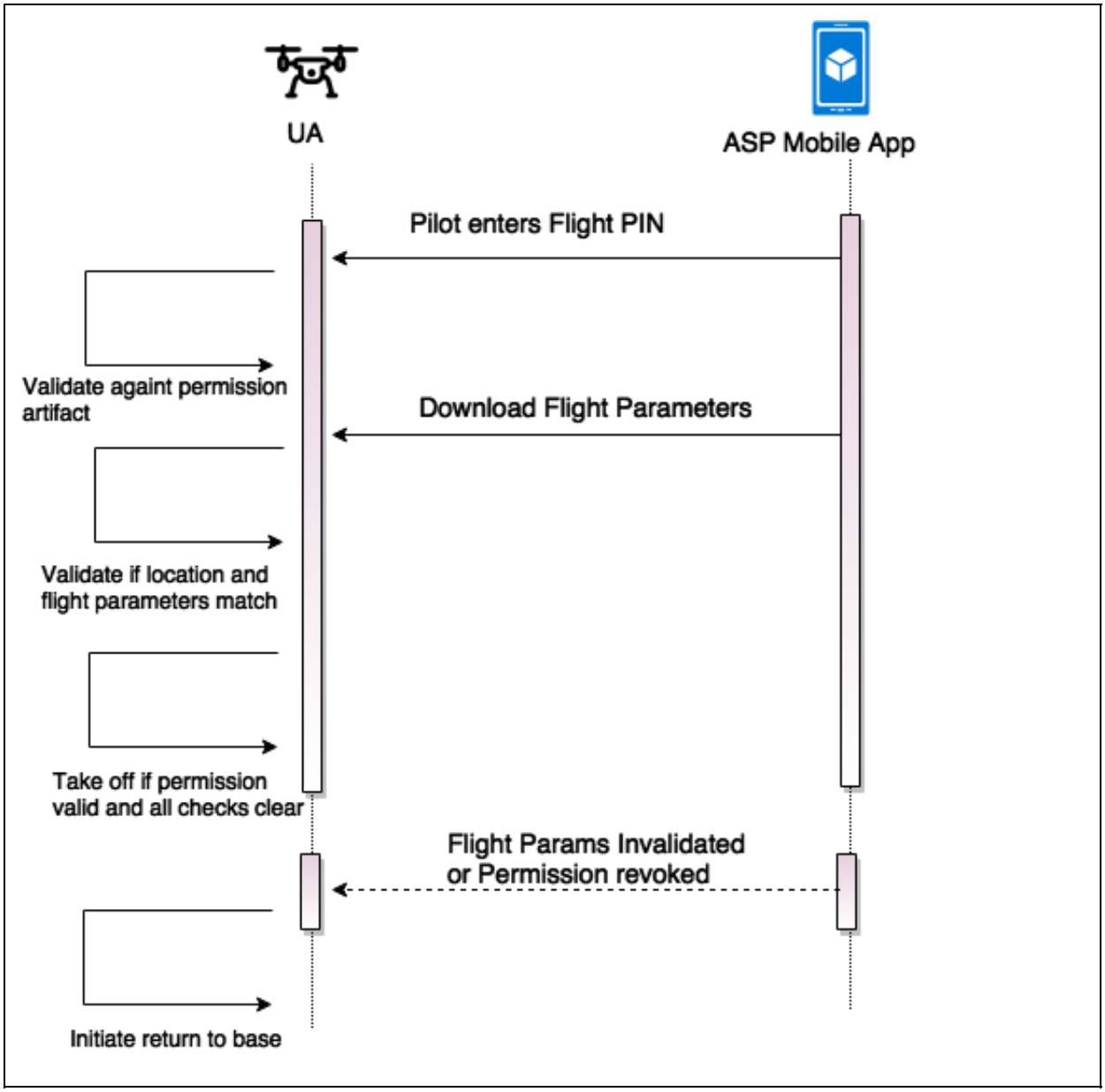
The approval request for a flight permission is designed in such a way so as to automate the approval as much as possible. The basic approval steps are:

1) The request has a valid transaction ID originating the from mobile app or the website
2) The Owner requesting the flight has registered with Digital Sky.
3) The pilot controlling the UA has a valid UAPL.
4) The UA has a valid UIN and is registered within Digital Sky by the manufacturer. There should also be a linking established between the UIN and the operator.
5) The Payload details are acceptable - basically the payload type and weight are acceptable
6) The Flight parameters - GPS coordinates (broad area), flight start and end time are acceptable. The UA should also use only permitted frequencies. This is required to ensure that there is no conflict with VIP traffic or national safety or other hazard conditions.
7) All required documents (liability, clearances, permits, identification, etc) are attached with the request.

If all parameters look good the flight is cleared for UTM approval. UTM system requires actual flight plan before the UA takes-off and works in a real-time manner. The actual flight plan is only provided at the time of flight after a survey by the pilot/operator on the date of flight. Only after UTM approval can a UA actually take off. The workflow is depicted below:


Request submitted from App/Website
UA Permission Request Message
Valid Transaction ID ?
No
Yes
Owner Registered & KYC Verified?
No
Yes
Valid Pilot (UAPL No)?
No
Yes
UIN should be registered with Digital Sky and there should be a link registered between the the operator and UA
Valid UA (UIN Valid)?
No
Yes
Payload Acceptable?
No
Yes
Flight Parameters Approved?
No
Yes
Required Documents Attached?
No
Yes
Cleared for UTM Approval
Permission Denied

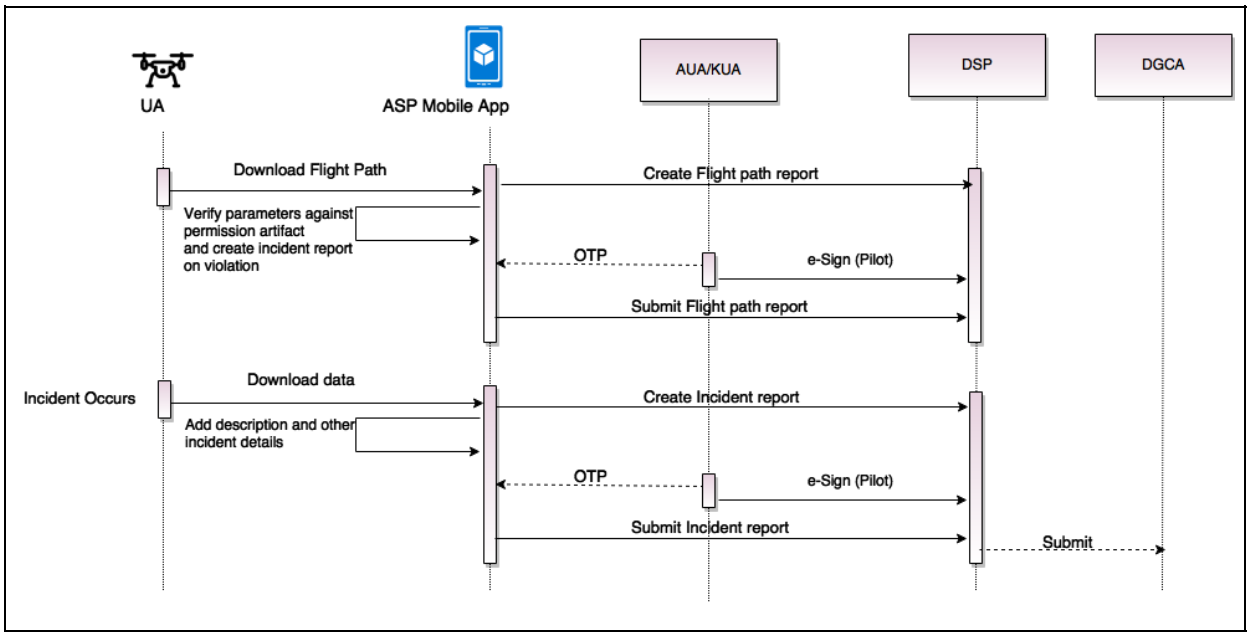
Permission Process - UA Take off



The takeoff and flight should work on the premise that the mobile app that is paired with the UA is where all the intelligence is built in. The basic workflow that should be supported is:

1. The pilot enters the Flight PIN that is generated after a successful permission request . The software on the UA should be able to check the cryptographically hashed PIN against what is keyed and take off only if it matches.
2. After the Flight PIN is verified , specific flight details are download to the UA. GPS coordinates and other flight parameters can be verified and for any violation, take off should be halted.
3. When there is a revocation of permission or invalidation or violation of parameters on which permission is requested, the UA must return to base and abort the flight.

Flight Logging and Incident Reporting



Pilots are required to submit flight path and incident reports once the flight is done. The process flow is explained below:

1. The pilot downloads the flight path details from the UA and creates a flight path report.
2. The pilot then packages the details of the flight, submits the report by e-Signing the artifact with Aadhaar.
3. An incident report is required where there is a deviation from what is approved in the permission artifact (like coordinates, height, timings etc). An incident report will have to be filed when an unforeseen event occurs - like a crash, property damage etc.
4. The incident report is e-signed by the pilot with Aadhaar and submitted to Digital Sky

Note: It is important to note that the permission workflow proposed is relevant for law enforcement agencies and not real-time air traffic control. Any such permission granted should qualify the operator/pilot to seek permission to use airspace (in that time frame and at that location) from UTM authority. UTM system requires actual flight plan before drone takes-off and works in a real-time manner.

Specifications

All request & response messages/artifacts in the Digital Sky ecosystem are cryptographically verifiable digital documents. All popular open formats like JSON and XML should be supported. The ASP can chose the format that works best for its application. DSPs should ideally support multiple formats and encodings.

Permission Artifact

The XML structure for the the permission artifact is shown below (XSD to be uploaded separately):

Permission Artifact

```
<UAPermission lastUpdated="" ttl="" txnId=""
permissionArtifactId="" pilotPinHash="">
    <Owner operatorID="">
        <Pilot uaplNo="" validTo=""/>
    </Owner>
    <FlightDetails>
        <UADetails uinNo=""/>
        <FlightPurpose shortDesc="" frequency="" />
        <PayloadDetails payLoadWeight="" payloadDetails=""/>
        <FlightParameters gpsCoordinates=""
            flightStartTime="" flightStartEnd="" frequenciesUsed=""
/>
    </FlightDetails>
    <DigitalSignature></DigitalSignature>
</UAPermission>
```

Element/Attribute	Description
UAPermissionRequest	Root element for the permission request
UAPermissionRequest->lastUpdated	Timestamp information
UAPermissionRequest-> ttl	Time to Live value in hours suggesting the caching and invalidation rules of the message
UAPermissionRequest->permissionArtifactId	The unique identifier for the permission artifact. This would be the reference for any re-issue or revocation
UAPermissionRequest->txnId	The transaction identifier generated by the digital sky mobile or web application.
UAPermissionRequest->pilotPinHash	The salted hashed PIN which would be verified against what the pilot enters before take off
Owner	Root element of the UA owner
Owner->operatorID	The owner's Operator ID obtained during registration
Pilot	Root element of the UA pilot
Pilot-> uaplNo	The Unmanned Aircraft Operator Pilot Registration number
Pilot->validTo	Expiry date of the UAPL
FlightDetails	Root element of the UA flight details
FlightDetails->UADetails->uinNo	The UIN of the UA for which the flight permissions are being requested
FlightPurpose->shortDesc	A short description for the purpose of the flight
FlightPurpose->frequency	Number of flights for the given flight parameters
Payload->payLoadWeight	The weight of the payload carried by the UA
Payload->payloadDetails	Any other details that can be furnished about the payload carried by the UA.
FlightParameters	The root element of for all details about the UA’s flight plan

FlightParameters->gpsCoordinates	The broad GPS coordinates (boundary) of the UA flight represented as a polygon of points
FlightParameters -> flightStartTime	The start time and date of the planned UA flight
FlightParameters->flightStartEnd	The end time and date of the planned UA flight
FlightParameters->frequenciesUsed	The planned frequencies to be used by the UA
DigitalSignature	The element containing the cryptographically verifiable signature of the applicant (this is generated from Aadhaar + Operator ID)

Permission Revocation

DGCA reserves the right to revoke any permission that is already issued. This may be on the grounds of national security or for any reason. The XML structure is explained below:

Permission Revocation Artifact

`<UAPermissionRevocation lastUpdated="" ttl="" permissionArtifactIdRef="" revocationArtifactId="">`
`  <RevocationDetails reason="" effectiveDate=""/>`
`  <DigitalSignature></DigitalSignature>`
`</UAPermissionRevocation>`

Element/Attribute	Description
UAPermissionRevocation	Root element for the revocation request
UAPermissionRevocation ->ttl	Time to Live value in hours suggesting the caching and invalidation rules of the message
UAPermissionRevocation ->permissionArtifactIdRef	The reference (unique id) of the previously generated permission request
UAPermissionRevocation ->revocationArtifactId	The unique identifier for the revocation artifact

UAPermissionRevocation ->lastUpdated	Timestamp information
RevocationDetails	Root element containing all information about the revocation
RevocationDetails ->reason	The specific reason provided for the revocation - a short description
RevocationDetails ->effectiveDate	The effective data time stamp post which the revocation is effective.
DigitalSignature	The element containing the cryptographically verifiable signature of the applicant (this is generated from Aadhaar + Operator ID)

Digital Sky Device Libraries for Manufacturers

These libraries will help manufacturers design/modify their UAS products in accordance with the Indian UAS regulations. The libraries are designed to only provide the regulatory information to the UAS firmware; the onus of using this information for strict policy enforcement is on the manufacturer and operator/pilot. These libraries will be open sourced in order to build trust among manufacturers. The regulatory authorities will be using various test cases to validate the strict policy enforcement by manufacturers, during the product verification tests.

For compatibility with all systems two libraries will be provided to manufacturers. The Onboard device library will be integrated with autopilot firmware and the Remote device library will be integrated with ground station software. Manufacturers are required to use both libraries.

Manufacturer is also required to provide communication infrastructure for data sharing between the two libraries.

Digital Sky Onboard Device Library:

This library is the point of contact for the autopilot firmware with Digital sky remote device library. The library will be designed with pure C, C++ implementation without any device specific components to ensure the compatibility.

Primary functions:

I. UAS Identification

II. Verifying authenticity of the permission artefact (Digitally signed certificate):
The permission artefacts will come with a digital signature (a form of Public key cryptography), encrypted using Digital Sky private encryption key. The library will use corresponding public keys (released by Digital Sky) to verify that the permission artefact is released by Digital Sky and has not been tampered with during transport. If the public key has to be changed (When Digital sky requests the manufacturers to change), the manufacturer has to release a firmware update to update the key. Manufacturer should not provide any option to the user to update the public keys used in the library.

III. Provide information of Time and Location bound restrictions to Autopilot firmware:
The library will provide geofence information in horizontal and vertical direction to the autopilot as provided in the permission artefact. The permission artefact also consists of the time period for which the artefact is valid.

IV. Collect flight data for security review:
The manufacturer is required to provide following data to the onboard device library. This data will be saved against each permission artefact.
    A. Date-time information from GPS data
    B. For each power cycle, the list of takeoff, land coordinates.
    C. Upon breach of the geofence, the timestamp and coordinates of the point where geofence is breached.
    D. Upon breach of the time limits, the additional time for which the UAS was in air should be provided to the library.

V. Send flight data to Digital Sky Remote Device Library:
Manufacturer has to provide the communication channel to the library so that the above logs can be shared with remote device library where they will be used for forming automated incident reports.

Digital Sky Onboard Device Library API Reference:

I. Update_permission:
Parameter 1: Permission_artefact
Parameter 2: UTC date-time (fetched from GPS data or any other source)
[provide permission artefact to the library.]
   A. This API should be called by autopilot firmware every time the UAS is powered on.
   B. Library will validate the permission artefact by deciphering the digital signature.
   C. Library will not store the permission artefacts in the memory, so manufacturer is responsible for providing relevant permission artefact to the library.
   D. Library will use timestamp provided by manufacturer and the certificate publish timestamp (marked by Digital Sky server when the certificate is created) to validate that the certificate is eligible at that time.

II. Get_Geofence_data:
Parameter 1: UTC date-time
[provides geofence information to the autopilot firmware using a valid certificate.]
   A. Library will verify the date-time stamp to reassert the time validity of the certificate.
   B. It is responsibility of manufacturer to ensure that the UAS follows the geofence restrictions. In case of geofence or time limit violations, the manufacturer should provide the details to the library.

III. Get_timelimit:
Parameter 1: UTC date-time
[Provides allowed time limit to autopilot using a valid certificate.]
   A. Library will verify the date-time stamp to reassert the time validity of the certificate.
   B. It is responsibility of manufacturer to ensure that the UAS is landed before the certificate time expires. In case of geofence or time limit violations, the manufacturer should provide the details to the library.

IV. Log_takeoff_location:
Parameter1: UTC date-time
Parameter2: GPS coordinates
[To provide takeoff event information to library for internal logging.]
   A. The manufacturer is responsible for calling this API immediately after every takeoff event.

B. The library will store and package all such events in one incident report against a permission artefact.

V. Log_Land_location:
Parameter1: UTC date-time
Parameter2: GPS coordinates
[To provide land event information to library for internal logging.]

A. The manufacturer is responsible for calling this API immediately after every land event.
B. The library will store and package all such events in one incident report against a permission artefact.

VI. Log_geofence_breach:
Parameter1: UTC date-time
Parameter2: GPS coordinates
[To provide geofence breach event information to library for internal logging.]

A. The manufacturer is responsible for calling this API immediately after every geofence breach incidence.
B. The manufacturer is responsible to force the UAS to return to home (as per the DGCA norms) on geofence breach incidence.
C. The library will store and package all such events in one incident report against a permission artefact.

VII. Log_timelimit_breach:
Parameter1: UTC date-time
Parameter2: time_overrun
[To provide time limit breach event information to library for internal logging.]

A. The manufacturer is responsible for calling this API immediately after every time limit overrun event.
B. The manufacturer is responsible to force the UAS to return to home (as per the DGCA norms) on geofence breach incidence.
C. The library will store and package all such events in incidence report against a permission artefact.

VIII. Get_individual_incidence_reports:
Parameter1: UTC date-time
[To get the individual incidence reports from the library for storage]

A. The library will prepare a incidence report after each land and Geofence breach event.
B. These incidence reports will be digitally signed by the library using a private key (belonging to the manufacturer only) to make sure that incident reports are not tampered with during the transport. It is manufacturer's responsibility to keep the private key secure. Also, manufacturer has to share

the public key with Digital Sky APIs where it will be used to verify the authenticity of the reports.

C. It is responsibility of the Manufacturer to collect these reports immediately after every land and Geofence breach event from library.

D. The manufacturer has to store all the incidence reports during the period of last 10 permission artefacts in a non-volatile memory storage. This storage should provide the 'write access' only to the autopilot firmware. The read access to the storage should be available in case of accidents. The manufacturer is required to provide the authorities with the specialized equipment required to read the incident reports from a crashed/damaged UAS's onboard storage.

IX. Bundle_incidence_reports:
Parameter1: UTC time-stamp
Parameter2: List_of_individual_incidence_reports_from_storage
Output1: Bundled_incident_report_with_digital signature.
[to bundle the signed incidence reports from storage into a single bundle (a bundle per permission artefact).]

A. After the time-period of a permission artefact is over, the user has to submit the incidence reports to Digital Sky APIs within 3 days. The manufacturer is required to provide all the incidence reports associated with a single permission artefact and pass them on to this API to get in return a signed bundle.

B. Manufacturer has to pass on this bundle to the Digital Sky remote device library APIs as an incidence report to be filed with Digital Sky APIs.

Digital Sky Remote Device Library:

This library will be the point of contact between Digital Sky Onboard Device library and Digital Sky APIs. It will include the workflow from seeking a flight permission to filing incident reports for the flights. The library will be developed for Android, iOS and Web platform. The workflow will be open to public and manufacturers of remote ground station applications may implement their own libraries following the specification. Manufacturer is required to provide all the necessary information to the remote library. However, it is user's responsibility to use the ground station softwares which follow the specification laid out by the authorities.

Primary functions:

I. The permission workflow:

II. Providing the permission artefact to the application:
The application has to interact with library and manufacturers communication infrastructure to pass on the permission artefact to the UAS.

III. Uploading incidence reports to Digital Sky APIs:
The library will upload the signed incidence report against a permission artefact to digital sky APIs.
It is Pilot's/Owner's/Operator's responsibility to upload the logs within the timeframe laid out by authorities.

Application developers responsibilities:

I. The permission workflow:

II. Passing on the certificate to UAS through Manufacturers communication infrastructure:
The application has to interact with library and manufacturers communication infrastructure to pass on the permission artefact to the UAS.

III. Passing on the incidence reports from Manufacturers communication infrastructure to the library:
The application has to interact with library and manufacturers communication infrastructure to pass on the permission artefact to the UAS.

Appendix

Section A: Test Structure for UAS Certification

Various test cases will be required to ensure that the manufacturer is following the regulations without physical examinations of the internal functionalities/firmware of the UAS. Such test cases will contain following primary tests:

- Fake and authentic signed permission artefact test.
- Individual incidence report storage test (including digital signature verification).
- Geofence breach incident report test.
- Time limit breach incident report test.
- Bundled incidence report test (including digital signature verification).
- RTH test in case of violation of permissions.
- Accidental storage data retrieval test with the specialized equipment.
- Overall no permission-no takeoff policy test.
- Any other test as defined by the regulator.

Haran's Notes

Technology Architecture and Framework for Registry

Registries are shared digital infrastructure onto which authorized issuing agencies publish digitally signed data about users, entities, or other assets/resources allowing consented and controlled access to other authorized service providers for digital verification and usage.

Features

- Flexible schema definition with complex structures
- Encryption and masking at attribute level
- Flexible data visibility classification

- Owner controlled consented access to service providers
- Digital signature for trusted data
- Open APIs for data publishing and registry access
- Flexible data lifecycle control at attribute level
- Extensibility hooks to build advanced features
- Ability to link registry entries within the same registry and across registries using different relationship types
- Extensible registry persistence model with support for graph based database, key-value stores, and blockchain
- Ability to attach rules for validations, lifecycle operations, and relationships

Use-Cases

- Large scale inter-dependent digital systems require access to machine readable and trusted registries of people, organizations and other assets/resources. These registries are typically setup for a purpose and made available to ecosystem for access.
- Such registries allow for seamless and automated integration of data and information at scale.

Technology Architecture and Framework for AirSpace Maps

Parking Area

Important questions to answer - determines the efficacy of the initiative:

1) Is the barrier to entry into the ecosystem much lower for operators and entrepreneurs?
2) How do we prevent a “License Raj” scenario if all operator registrations are to be blessed by DGCA?
3) Can operator registration be a scalable process without any delay in issue of subsequent flight permissions?
4) What is the format of Operator ID? How do we uniquely identify operators?
5) How is QR Code for UAs scanned? Can it be mandated for the manufacturers to embed (just like all cars have a VIN barcode)?
6) How can property owners provide permissions to operators in a seamless way? This process should not become a bottleneck?
7) What do we mean by Privacy-first? What aspects of the architecture deal with privacy?